

MICROHOBBY

AMSTRAD

Semanal

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

AÑO I N.º 14

150 Ptas.

Canarias 160 ptas.

FILER:

crea tus archivos
de datos en
cinta/disco

¡Quinielas!

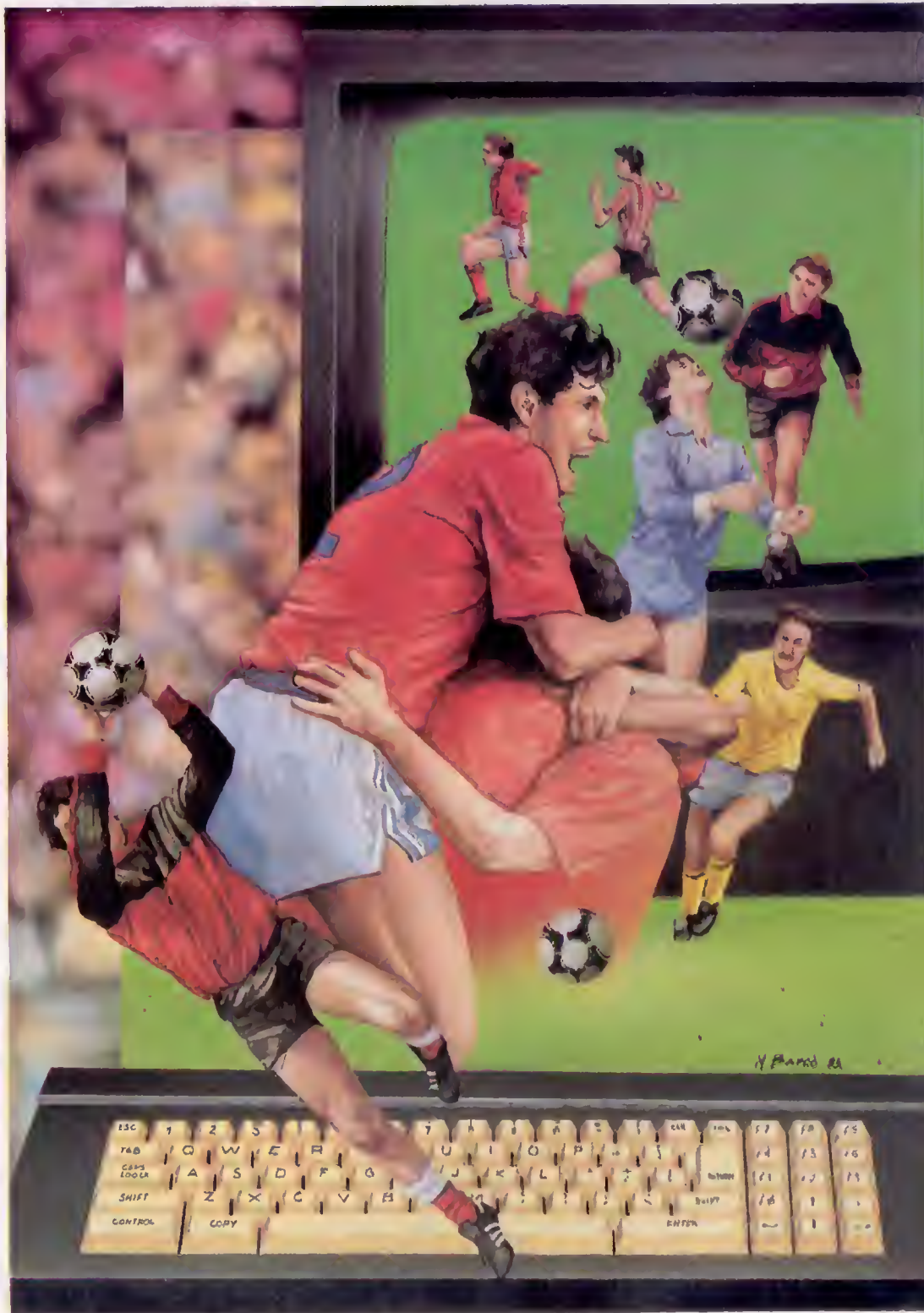
Hazte millonario
con tu Amstrad:
con quinielón
podrás acertar
hasta 14
resultados

OPERACIONES MATEMATICAS EN EL AMSTRAD

RUTINAS DE SPRITES E IMPRESION DE CARACTERES EN PANTALLA

SOFTWARE

DYNAMITE DAN,
primer juego comercial
en España con pokes
de vidas infinitas



UN "MILLON" EN JUEGO

Gráficos a color, rapidez de acción y sonido dan a este juego la sensación de una máquina de millón real.

PVP.
1.900 (cinta)
2.900 (disco)



464
664
6128
AMSTRAD

ACE

ACTIVIDADES COMERCIALES ELECTRÓNICAS, S.A.
Torreón, 110. Tel. 325 10 50* 09015 Barcelona. Telex 53133 ACEE E

YA DISPONIBLE EN

El Corte Inglés

... Y EN TODAS LAS
TIENDAS ESPECIALIZADAS

TARJETA DE SUSCRIPCION AHORRO/REGALO

(Si lo prefieres suscríbete por teléfono (91) 733 50 12 (91) 733 50 16)

Oferta especial para recibir en tu domicilio todo un año la revista semanal

Microhobby AMSTRAD

con un descuento de 1.600 ptas., y el regalo de una cinta original por valor de 2.100 ptas.!

FECHA LIMITE DE RESPUESTA:

30 DE NOVIEMBRE DE 1985

¡RESPONDE HOY MISMO!

¡SEÑALA EL NOMBRE DE LA CINTA QUE
PREFIERES Y LA RECIBIRAS EN TU CASA,
GRATIS, A VUELTA DE CORREO!

¡Deseo suscribirme a **Microhobby AMSTRAD** durante un año por sólo **5.900 ptas.**, lo que me supone un ahorro de **1.600 ptas.**

El primer número que deseo recibir es el _____

Envíame **GRATIS** la cinta de programas que le indico con una (X)

☐ **Beach Head**

☐ **Combat Lynx**

☐ **D.T. Decathlon**

NOMBRE _____ EDAD _____

APELLIDOS _____

DOMICILIO _____

CIUDAD _____ PROVINCIA _____

C. POSTAL _____ TELEFONO _____ PROFESION _____

Marco con una (X) en el casillero correspondiente la forma de pago que más me conviene.

☐ Talón bancario adjunto a nombre HOBBY PRESS, S. A. ☐ Giro Postal N.º _____ ☐ Contra reembolso del
primer número, junto a la cassette regalo. ☐ VISA N.º _____ ☐ MASTER CHARGE N.º _____

Fecha de caducidad de la tarjeta _____

Firma: _____



8

n dinamita
er destructivo Rayo



Serie oro 12

El programa quinielas que presentamos en este número muy bien podría calificarse de excepcional. Nos permite hacer toda tipo de apuestas desarrollando las distintas combinaciones según los métodos considerados mejores por la «teoría» quinielística.

Las probabilidades de acertar 14 son pocas, pero tal vez el programa sirva de ayuda a alguien y le permita investigar la estrategia más conveniente.



20 ProgramAcción

Nunca está de más mejorar la apariencia estética de nuestros programas, añadiéndoles belleza y originalidad.

En este sentido presentamos dos nuevos comandos del **Amstrad Basic**, creados haciendo uso del sistema **RSX**, que permiten girar caracteres a derecha e izquierda de 90 en 90 grados.

Código máquina 28

Damos un paso de gigante en el estudio de los gráficas y, por tanto, de la forma en la que el **Amstrad** organiza la memoria de pantalla. También, las primeras rutinas de impresión de gráficos en código máquina.



Análisis 33

Un pequeño programa muy bien construido nos muestra la técnica de los ficheros de texto en cinta/disco. Como escribir, leer en ellos, abrirlos y cerrarlos.

Presidente

Maria Andrina

Consejero Delegado

José I. Gámez-Centurión

Jefe de Publicidad

Cancha Gutiérrez

Publicidad Barcelona

José Galán Cartes

Tel: (93) 303 10 22/313 71 62

Secretaria de Dirección

Marisa Cagarra

Suscripciones

M.ª Rosa Ganzález

M.ª del Mar Calzada

Redacción, Administración y Publicidad

La Granja, s/n

Poligona Industrial de Alcobendas

Tel.: 654 32 11

Telex: 49 480 HOPR

Dto. Circulación

Carlos Perapadre

Distribución

Caedis, S. A. Valencia, 245

Barcelona

Imprime

ROTEDEC, S. A. Ctra. de Irún.

Km. 12,450 (MADRID)

Fotocomposición

Navacamp, S.A.

Nicolás Marales, 38-40

Fotomecánica

GROF

Ezequiel Salana, 16

Depósito Legal:

M-28468-1985

Derechos exclusivos
de la revista

**COMPUTING with
the AMSTRAD**

Representante para Argentina, Chile,
Uruguay y Paraguay, Cia.
Americana de Ediciones, S.R.L. Sud
América 1.532. Tel.: 21 24 64. 1209
BUENOS AIRES (Argentina).

M. H. AMSTRAD no se hace
necesariamente solidaria de las
opiniones vertidas por sus
colaboradores en los artículos
firmados. Reservadas todas las
derechos.

Se solicita control OJD

Franqueo
Postal

Grat

HOBBY PRESS, S.A.

Apartado de Correos

n.º **54.062** (Apartados Altos)

MADRID

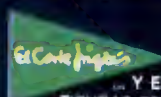
¡¡SUSCRIBETE A Microhobby AMSTRAD Y AHORRA 1.600 PTAS.!!



ACE

ACTIVIDADES COMERCIALES ELECTRONICAS, S.A.
Tarragona, 110. Tel. 325 10 50* 09015 Barcelona. Telex 30133 ACEEE

YA DISPONIBLE EN



Y EN TODAS LAS
TIENDAS ESPECIALIZADAS

Director Editorial
José I. Gómez-Centurión

Director Ejecutivo

Víctor Prieto

Subdirector

José María Díaz

Redactora Jefe

Marta García

Diseño

José Flores

Colaboradores

Francisco Portalo

Pedro Sudón

Miguel Sepúlveda

Francisco Martín

Jesús Alonso

Pedro S. Pérez

Amalia Gómez

Juan J. Martínez

Secretaría Redacción

Carmen Santamaría

Fotografía

Carlos Condel

Javier Martínez

Portada

Manuel Barco

Ilustradores

J. Igual, J. Pons, F. L. Frontán,

J. Septien, Pejo, J. J. Mora,

Luigi Pérez

Edita

HOBBY PRESS S.A.

Presidente

María Andino

Consejero Delegado

José I. Gómez-Centurión

Jefe de Publicidad

Concha Gutiérrez

Publicidad Barcelona

José Galán Cartes

Tel: (93) 303 10 22/313 71 62

Secretaría de Dirección

Marisa Cogorro

Suscripciones

M.ª Roso González

M.ª del Mar Colzoda

Redacción, Administración y Publicidad

La Granja, s/n

Polígono Industrial de Alcubendas

Tel.: 654 32 11

Telex: 49 480 HOPR

Dto. Circulación

Carlos Peropadre

Distribución

Caedis, S. A. Valencia, 245

Barcelona

Imprime

ROTEDEC, S. A. Crta. de Irún.

Km. 12,450 (MADRID)

Fotocomposición

Novocomp, S.A.

Nicolás Morales, 38-40

Fotomecánica

GROF

Ezequiel Solano, 16

Depósito Legal:

M-28468-1985

Derechos exclusivos de la revista

COMPUTING with the AMSTRAD.

Representante para Argentina, Chile, Uruguay y Paraguay, Cia. Americana de Ediciones, S.R.L. Sud América 1.532. Tel.: 21 24 64. 1209 BUENOS AIRES (Argentina).

M. H. AMSTRAD no se hace necesariamente solidaria de las opiniones vertidas por sus colaboradores en los artículos firmados. Reservados todos los derechos.

Se solicita a control OJD

Año 1 • Número 14 • 3 al 9 de Diciembre de 1985
150 ptas. (sobretasa Canarias, 10 ptas.)

8 Primeros pasos

Además de las cadenas alfanuméricas, de indudable utilidad, es necesario conocer como se manejan los números y los cálculos relacionados con ellos en el **Amstrad**. Primeros pasos estudia el tema y propone el atento examen de un interesante programa de demostración.

Serie oro 12

El programa quinielas que presentamos en este número muy bien podría calificarse de excepcional. Nos permite hacer todo tipo de apuestas desarrollando las distintas combinaciones según los métodos considerados mejores por la «teoría» quinielistica.

Las probabilidades de acertar 14 son pocas, pero tal vez el programa sirva de ayuda a alguien y le permita investigar la estrategia más conveniente.



Análisis 33

Un pequeño programa muy bien construido nos muestra la técnica de los ficheros de texto en cinta/disco. Como escribir, leer en ellos, abrirlos y cerrarlos.

Mr Joystick 18

¿Conseguirá el superogente Don dinamita apoderarse de los planos del superdestructoro Rayo Devastador?



20 ProgramAcción

Nunca está de más mejorar la apariencia estética de nuestros programas, añadiéndoles belleza y originalidad.

En este sentido presentamos dos nuevos comandos del **Amstrad Basic**, creados haciendo uso del sistema RSX, que permiten girar caracteres a derecha e izquierda de 90 en 90 grados.

Código máquina 28

Damos un paso de gigante en el estudio de los gráficos y, por tanto, de la forma en la que el **Amstrad** organiza la memoria de pantalla. También, las primeras rutinas de impresión de gráficos en código máquina.



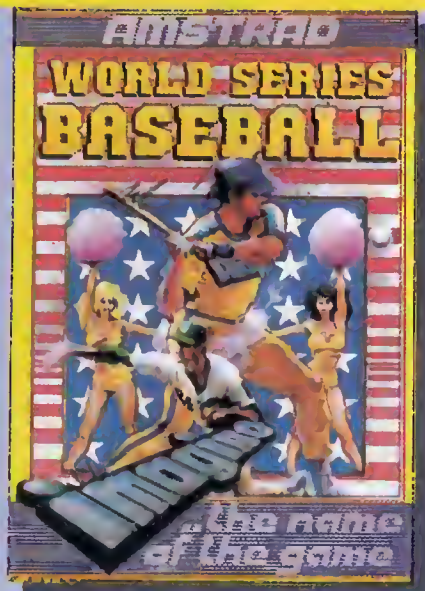
¡NUEVO!

SIEMPRE LOS PRIMEROS EN TENER LO ULTIMO

círculo de soft

MICROAMIGO S.A.

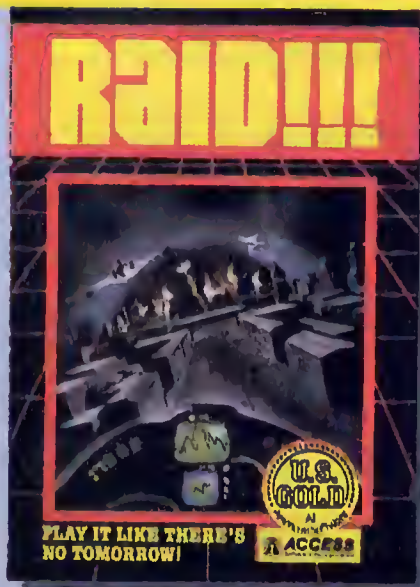
P.º de la Castellana, 268, 3.º C. 28046-MADRID.
Tel.: (91) 733 25 00



BASEBALL

Impresionante simulación en 3ª dimensiones. Se puede competir contra el ordenador o contra otro jugador. No es necesario conocer el beisbol. Hay un modo de demostración. Pantallas gigantes para ver de cerca la acción.

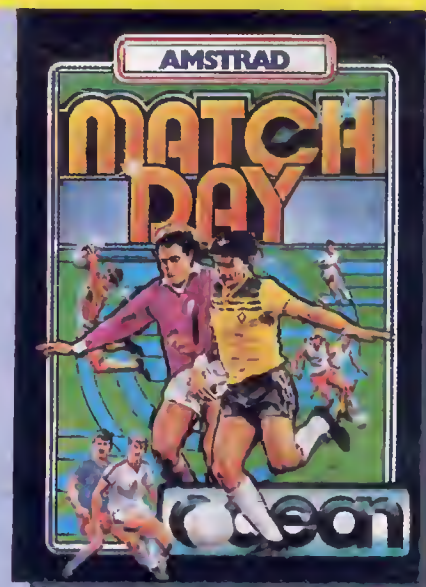
P.V.P.: 2.100 ptas.
Precio C. de Soft: 1.890 ptas.



RAID

Defiéndete con tu escuadrilla de aviones del ataque nuclear que han lanzado sobre ti. Tu viaje estará lleno de peligros hasta que llegues a las bases de lanzamiento de misiles enemigas. Tendrás que destruirlos para salvar a tu país de una catástrofe nuclear. Gráficos y acción sensacionales.

P.V.P.: 2.300 ptas.
Precio C. de Soft: 2.070 ptas.



MATCH DAY

¡Ahora para Amstrad! No se trata de un juego de fútbol cualquiera. Fantástica acción en 3 dimensiones y animación total que dan vida al fútbol. Quedarás maravillado con el control del balón y desarrollarás tu destreza y técnica jugando contra otro jugador o contra el ordenador.

P.V.P.: 2.300 ptas.
Precio C. de Soft: 2.070 ptas.

¡¡¡...Y LOS TRES PROGRAMAS POR SOLO 5.400 PTAS!!!

¡HAZTE HOY MISMO SOCIO DEL CIRCULO DE SOFT! Además de poder adquirir tus programas al mejor precio, recibirás información de forma periódica y gratuita, del mejor software que aparezca en el mercado.

¿QUE HAY QUE HACER PARA SER SOCIO DEL CIRCULO DE SOFT? Así de fácil: envíanos por correo tu nombre, dirección y modelo de ordenador, o bien, pide por teléfono o por correo tu primer programa. ¡Y entrarás a formar parte del CIRCULO DE SOFT de forma inmediata!

☐ Si, quiero ser SOCIO desde hoy mismo del CIRCULO DE SOFT y recibir periódicamente información de novedades de software, así como beneficiarme desde hoy mismo de los precios reducidos reservados a los SOCIOS y de sus Ofertas Especiales. El ser SOCIO no me obliga a compra alguna.

Si prefieres formalizar tu compra por teléfono puedes hacerlo llamando al (91) 733 25 00. ¡NO SE COBRAN LOS GASTOS DE ENVIO POR CORREO!!

TITULO _____		P.V.P. _____	ORDENADOR _____
_____		_____	_____
_____		_____	_____
_____		_____	_____

☐ Contrarreembolso
 ☐ Giro Postal
 ☐ Talón adjunto a Microamigo, S.A.
 ☐ Tarjeta VISA n.º _____ Fecha caducidad _____

Nombre _____ Apellidos _____ Edad _____

Domicilio _____ Teléfono _____

Localidad _____ C.P. _____ Provincia _____



UNIX PARA EL QL

Este sistema operativo es, por definición, multiusuario y multiárea, posee una estructura jerárquica de ficheros en disco, con subdirectorios, y un sistema de movimiento de datos a través de «pipelines», pudiendo redirigir la salida de un programa a otro o a cualquier dispositivo físico del ordenador, de una forma muy sencilla.

Los lectores se percatarán de que estamos describiendo un microUnix. Su versatilidad y potencia son enormes, siendo además compatible con el Unix Standard a nivel de código fuente del lenguaje C.

El QL ve incrementada su ROM a 144 Kbytes por la adición del OS9.

La casa británica inglesa Cumana, conocida entre otras cosas por las unidades de disco que fabrica, ha desarrollado para el Sinclair QL una versión de sistema operativo OS9, hasta ahora implementado solamente en los ordenadores Dragon dentro del campo de los micros domésticos.

Primera plana



NUEVO MODELO DE IBM

IBM ha lanzado una configuración superior de su último modelo, el IBM AT, el ordenador personal más potente del mundo (según IBM).

La nueva configuración posee 512 K de RAM, un Floppy de 1,2 megabytes y un disco duro de 30 millones de caracteres de almacenamiento.

Al parecer, una de las razones de la aparición del nuevo modelo se debe a la próxima aparición, valga la redundancia, de otra máquina basada en el 80386 de Intel, el no va más de los procesadores actuales.

Y la otra razón, dicha en temerosos susurros, es que el disco de 20 millones del AT tenía un porcentaje de fallos anormalmente alto. IBM nunca reconoció este hecho, pero en USA lanzó un programa de alcance nacional para identificar y sustituir las placas de los controladores de disco sospechosas.

ORDENADORES VIVOS

Comparado con los ordenadores de hace 10 años, el **Ams-trad** que usted tiene en su casa les da «mil vueltas», porque es mucho más potente, pequeño y barato.

Estas mejoras se deben a muchos factores, naturalmente, resumidos todos ellos en el tópico «avance de la tecnología».

Sin embargo, dentro de todo este cúmulo, existe uno de importancia capital, responsable en gran parte de la presencia cada vez más amplia del ordenador en nuestra vida: la microminiaturización.

Resulta que cuanto más pequeño es un chip, más barato resulta, y la máquina que lo incorpora deviene más potente por el simple hecho de que le caben más.

Estas razones tienen el suficiente peso económico para que se investigue sin cesar en cómo fabricar chips cada vez más pequeños, con una mayor escala de integración, que se dice.

Tanto es así, que en un futuro cercano se alcanzará el límite. Simplemente no podrán fabricarse chips más pequeños (aunque en ese momento los de un mega serán cosa corriente).

No es de extrañar que la gente esté empezando a pensar en materiales e incluso tecnologías alternativas a la electrónica. Las opciones son

muchas, pero tal vez la más atractiva y revolucionaria sería construir ordenadores basados en material biológico, en el carbono en lugar del silicio.

Si esto pudiera conseguirse, en el mismo espacio físico la capacidad de proceso y almacenamiento se vería multiplicada por un factor de miles de veces, ya que los «transmisores» de información, serían las propias moléculas orgánicas, muchísimo más pequeñas que cualquier chip.

Y todavía hay más: un importante sector de la comunidad científica que investiga el tema ve la llegada de los ordenadores biológicos como un paso más de la evolución, que hasta ahora culminaba en nosotros. Incluso se plantean la posibilidad de la unión directa, la simbiosis, hombre ordenador.

De momento, todas estas inquietantes y atractivas posibilidades entran dentro del campo de la ciencia ficción (¿a Dios gracias?), pero nombres como IBM, Hitachi y una gran cantidad de universidades con los mejores cerebros del mundo van a hacer todo lo posible para convertir las en realidad en el mínimo tiempo posible.

¿Dejarán los ordenadores de ser simplemente una herramienta estúpida para llegar a ser parte de nosotros mismos? ¿Para mejorarnos?

¿Cuánto tiempo la palabra «humano» se ajustará a nuestros estándares de hoy?

ROBOTS ESCULTORES

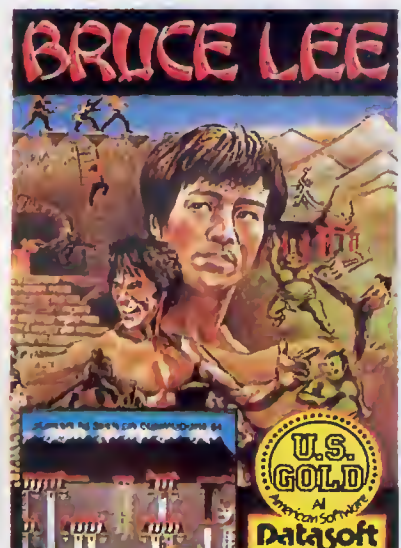
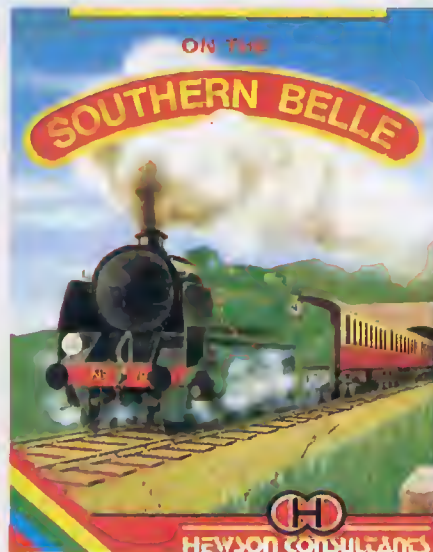
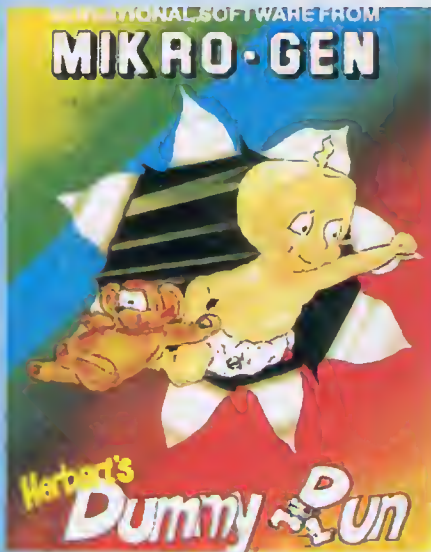
Renault está utilizando en Francia brozos robot controlados por ordenador para esculpir en plástico los modelos de los prototipos de sus coches. El tiempo que toma este proceso se ha reducido a 6 semanas, mediante la ayuda de este completo sistema CAD.

El modelo se esculpe aprovechando el output de otro programa que dibuja el coche en dos dimensiones, el cual se traslada a tres dimensiones automáticamente.



PREPA

Dis



SI BUSCAS LO MEJOR **ERBE** Software LO TIENE

RA Las MALETAS A

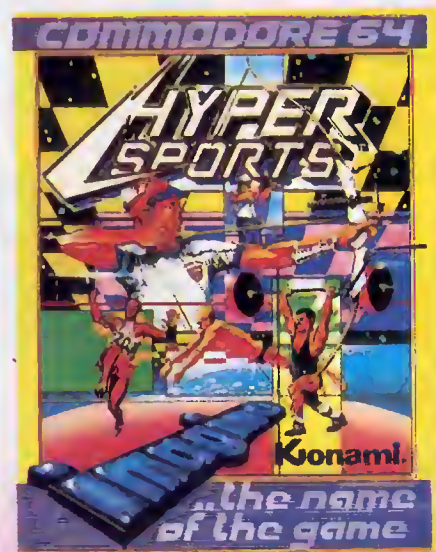
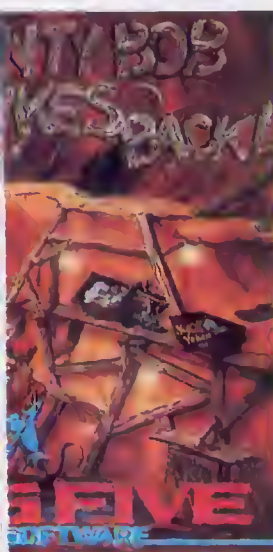
ney World!



Software

TE INVITA

COMO LO OYES. EN CADA UNO DE NUESTROS JUEGOS ENCONTRARAS UN CUPON. MANDANOSLO DEBIDAMENTE RELLENADO Y PARTICIPARAS EN EL SORTEO QUE ENTRE TODOS LOS CUPONES RECIBIDOS HAREMOS EL 15 DE ENERO. ¡IMAGINATE! NADA MENOS QUE VIAJE Y ESTANCIA PARA 2 PERSONAS DURANTE 1 SEMANA EN MIAMI, EN DISNEY WORLD.



STA. ENGRACIA, 17 - 28010 MADRID. TEL. 447 34 10

NUMEROS

Hemos comentado en esta sección más de una vez que los ordenadores manejan exclusivamente números, al menos internamente.

El Amstrad no es una excepción y, por ello, vamos a revisar un poco cómo se tratan los números y qué puede hacerse con ellos.

Javier Igual

Lo más básico es comenzar con las cuatro operaciones matemáticas básicas: suma, resta, multiplicación y división.

Naturalmente que sabemos que todos dominan la aritmética sencilla, todos... menos el **Amstrad**, y ello puede crearnos algunos serios problemas.

Por ejemplo:

`PRINT 5+3`

obviamente imprimirá un hermoso 8 en la pantalla, pero:

`PRINT 5+3*8`

que resultará, 64 ó 29?

La ambigüedad de las expresiones

Como podéis observar, estamos en dificultades. Una persona ajena que leyera la expresión anterior dudaría sobre la intención del que la escribía, en el sentido de que se pretenda conseguir un resultado u otro.

Los ordenadores no pueden dudar, porque si lo hacen, se bloquean y dejan de funcionar. Para una máquina una cosa es 0 ó 1, cierta o falsa.

Esto implica la existencia de una serie de reglas incorporadas al Basic para poder manipular expresiones numéricas de una forma unívoca, sin lugar a ambigüedades.

Su nombre: Regla de la Precedencia de Operadores.

Esto quiere decir, sencillamente, que por acuerdo asignamos un orden de ejecución a los diferentes símbolos que operan sobre los números, por ejemplo, si la máquina se encuentra un signo «+» y un signo «*», podríamos decidir que primero se evalúen los miembros de la expresión que lo contengan; en el caso anterior de:

`PRINT 5+3*8`

el resultado sería 29. Si hubiéramos decidido que la suma tiene mayor prioridad, obtendríamos 64, un resultado tan diferente que la necesidad de unas reglas se hace patente.

La prioridad es la siguiente, de mayor a menor:

- Potenciación
- Multiplicación y división
- Suma y resta

Las operaciones que poseen la misma prioridad se evalúan de izquierda a derecha, por lo que el resultado «correcto» del ejemplo anterior es 29, no 64.

Bien, ¿y si yo quiero que la operación se realice justo de la forma «errónea»?

Todos los lenguajes de programación han previsto que en un momento dado interesa saltarse las reglas, y el método escogido para indicárselo al ordenador es casi siempre el mismo, derivado de la forma matemática normal de operar.

Gracias a los paréntesis

Sí, es como te enseñaron en el colegio, mediante el uso de paréntesis.

Obtendremos 64 si escribimos:

`PRINT (5+3)*8`

Esta es la regla general para las 4 operaciones básicas, pero en el caso de la multiplicación y división el asunto está un poco menos claro debido a la propiedad conmutativa de la multiplicación.

En efecto, MATEMATICAMENTE es lo mismo calcular:


`PRINT 5*8/5`

que

`PRINT (5*8)/5`

o bien

`PRINT 5*(8/5)`



Primeros pasos

aunque aún en este caso, las reglas de precedencia se siguen al pie de la letra. Sin embargo, las propiedades matemáticas de los operadores le juegan una mala pasada a la claridad.

Un poco más complejo es el tema de «juguetear» con números fraccionarios. **¿Podría usted predecir los posibles resultados de:**

$$5/4 + 5 * 18/9 - 27$$

o algo parecido?

Espero que no, porque yo desde luego no puedo. Si es capaz de ello, mis más sinceras felicitaciones. Para aquellos que tenemos una mente menos rápida y un ojo menos agudo, la regla de oro es la siguiente: en la duda, utilice los paréntesis, que para eso están.

Es la única forma segura de no equivocarse y obtener resultados disparatados en su programa debido a errores que, como éstos, son difíciles de depurar.

Segunda regla de oro: deje que su ordenador haga la mayor cantidad de trabajo posible. Usted sólo siéntese y disfrute.

La exponenciación

A nivel de operaciones básicas, nos queda por hablar solamente de la exponenciación, esto es, la operación matemática que toma un número, llamado BASE y lo multiplica tantas veces por sí mismo como indique un segundo número, llamado exponente. Se representa así: A^B , en donde A es la base.

El estudio de esta operación implica interesantes posibilidades de aprender mucho más acerca del tratamiento de los números y de los límites de cálculo del **Amstrad** y de los ordenadores en general. Atención a lo que sigue.

Lo primero que uno podría plantearse es hasta dónde llega la tan careada capacidad de cálculo de los ordenadores, del **Amstrad**.

En efecto, por muy potente que sea un micro, el rango de números que puede calcular debe tener un límite, fundamentalmente porque los números, al representarse internamente en el ordenador, ocupan memoria.

Cada máquina tiene asignado un número fijo de bytes para representar un número y ahí se alcanza el techo de cálculo.

¿Cómo averiguarlo? Bien, el método de «la cuenta de la vieja» nunca falla, así que podríamos ordenarle al **Amstrad** que imprimiera 3 elevado a 1.000, por ejemplo.

Probablemente, el número astronómico que resulta sobrepasaría la capacidad de cálculo del ordenador, con lo que un oportuno mensaje de error nos forzaría a reintentar con otro número más pequeño.

Esta manera de proceder no es precisamente muy astuta.

Tomarle el pelo al Amstrad

Nosotros, que se supone que pensamos, podemos aprovecharnos de la estupidez del micro, para que nos diga de una sola vez el rango de números que puede manejar.

Los matemáticos llevan diciendo hace mucho que un número dividido por cero es «infinito», es decir, primero, que si dividimos algo por nada, como resultado debemos obtener una cifra gordísima y segundo, al no saber exactamente cuál es, la llamamos «infinito» y listo.

Para una persona, este razonamiento resulta claro, pero para el **Amstrad** «infinito» quiere decir el máximo número que puede representar.

Por tanto, la orden:

```
PRINT 3/0
```

mostrará en pantalla el techo de cálculo del ordenador.

Este número es del orden de 10 elevado a 38, es decir, un 1 seguido de 38 ceros (*mil millones es un 1 seguido de 9 ceros*); una cifra inimaginable.

Bueno, el primer punto resuelto. El cálculo del techo inferior es igual de simple:

```
PRINT (3/0)↑-1
```

o sea, le decimos al **Amstrad** que divida un 1 entre (3/0) [«infinito»]. El resultado será «menos infinito», el número más pequeño que el ordenador puede manipular. En este caso,

```
10 REM PROGRAMA NUMERO 1
20 ON ERROR GOTO 280
30 :
40 CLS
50 overflow.error=6:ok=0:flag.error=0
60 :
70 WHILE NOT ok
80 :
90 GOSUB 200
100 :
110 PRINT "Calculando"
120 res=base^ex
130 :
140 GOSUB 240
150 :
160 PRINT base;"elevado a";ex;"vale ";res
170 WEND
180 END
190 :
200 INPUT "Base: ",base
210 INPUT "Exponente: ",ex
220 RETURN
230 :
240 IF flag.error THEN PRINT "Ha habido un overflow":PRINT "El maximo exponente posible es:";ex
250 flag.error=0
260 RETURN
270 :
280 IF ERR<>overflow.error THEN RESUME NEXT
290 flag.error=-1
300 ex=ex-1
310 RESUME
320 :
```

aparece en la pantalla una cifra del orden de 10 elevado a -39.

¿Y qué pasa cuando nos salimos del rango permitido?

Fuera de rango

Entonces, el **Amstrad** nos obsequia con un lacónico mensaje de error: «overflow», que traducido al español significa «sobrepasamiento» y el programa se detiene.

Lejos de representar un inconveniente en los programas en los que usemos cálculos numéricos, podemos aprovechar este error de una forma muy conveniente, como por ejemplo en el programa número 1.

Este programa calcula el máximo exponente al cual puede elevarse un número para no salirse del rango de cálculo del **Amstrad**.

El procedimiento es muy simple, ya que sólo pueden suceder dos cosas:

a) El resultado de elevar un número (*la base*) al otro (*el exponente*) cae dentro del rango de cálculo del micro. Estupendo, lo imprimimos y punto.

b) El resultado cae fuera de rango: se produce un error overflow, lo detectamos mediante la sentencia 20 del programa, variamos el exponente y repetimos el intento.

Por tanto, el programa 1 consta de cuatro partes:

1) bucle principal (sin fin): líneas 70-170;

2) subrutina de entrada de datos: líneas 200-220;

3) si ha ocurrido un error de overflow, se imprime un mensaje en pantalla, indicando el valor máxima posible del exponente: líneas 240-260;

4) subrutina de captura de errores: líneas 280-310.

Manejo de errores y cálculo numérico

La parte 4 del programa es el meollo del asunto; en efecto, sigamos las pasas del ordenador cuando se produce un error. La máquina salta, por efecto de la línea 20, a la rutina de la línea 280.

Esta línea comprueba que el valor numérico del error ocurrida sea igual al de la variable «**overflow.error**» (cada error posee un código numérico asociado; el 6 corresponde a overflow; ver capítulo 7 del manual). Si no lo es, nadie nos ha invitado a esa fiesta, por lo que optamos por ignorarlo mediante la orden:

RESUME NEXT

que fuerza al Basic a «pasar» de esa línea y ejecutar la siguiente. Aquí no ha pasado nada.

Si efectivamente el error detectado es el número 6, en la línea 290 la variable «**flag.error**» se pone a 1 para indicar a la rutina de la línea 240 que imprima un mensaje cuando le toque. Las líneas 300 y 310 son la auténtica clave del programa.

Cuando la rutina de error llega a esta línea, el valor de la variable «**ex**», que representa al exponente, se decrementa en una unidad, y en la línea 310 la orden:

RESUME

obliga al **Amstrad** a continuar la ejecución EN LA MISMA LINEA DE PROGRAMA DONDE SE PRODUJO EL ERROR, es decir, en la línea 120.

Esto implica que el ordenador repetirá el cálculo de nueva, pero con el exponente decrementada en una unidad, y lo hará una y otra vez hasta que en la línea 120 no se produzca error alguna, o sea, cuando el exponente sea el máximo al que podemos elevar la base sin salirnos del techa numérica del micro.

Interesante, ¿verdad?

Sin embargo, el programa no es perfecta. Como se observará al jugar con él un rato, con los números enteros o reales positivos todo marcha a pedir de boca, pero con los negativos al usar un exponente muy bajo el resultado final es cero.

El **Amstrad** se niega a considerar un error el overflow por «debajo». ¿Alguien podría explicar por qué?

Y, sobre todo, ¿alguien podría explicar por qué a nosotros?

Le quedaríamos muy agradecidas.



NOVEDAD

Solamente lo mejor



2.300 ptas.



1.700 ptas.



1.900 ptas.



1.900 ptas.



1.900 ptas.



2.100 ptas.



2.100 ptas.

PIDELOS POR CORREO.

COMPTIQUE

Embajadores N.º 90
Madrid 28012. Telf.: 227 09 80

QUINIELON

Quiniela siempre suena a dinero, a idilio con la suerte. Los ordenadores no pueden calcular rigurosamente todas las innumerables posibilidades de un boleto quinielístico, pero sí pueden seleccionar uno, en base de una serie de criterios (estrategia) elegida por el programador, y, además, hacer mucho más sencilla el tedioso protocolo del relleno de los boletos.

Así que, adelante: los 14 duermen en la RAM de tu Amstrad.

Realizado por el lector:
J. ARACIL XARRIE (Barcelona)

COMPATIBLE
CPC 464
CPC 664
CPC 6128

Como su propio nombre indica, este programa es una excelente aplicación para el desarrollo de nuestros boletos quinielísticos.

El autor señala en su extenso y detallado comentario del funcionamiento del programa que ha escogido un método, un tanto por «decreto», para clasificar lógicamente los partidos, de forma que la posibilidad de obtener un 1, X ó 2 en un determinado encuentro es distinta.

Lleno de modestia, Javier llama a su programa «cuasialeatorio»; ustedes decidirán sobre la acertado o no de la denominación.

En el peor de los casos, lo que no se puede discutir es que el programa simplifica extraordinariamente el tedioso proceso de rellenar boletos.

DESCRIPCION

El programa permite realizar apuestas quinielísticas, de una sola columna o de ocho columnas, limitando a voluntad del usuario el número de dases.

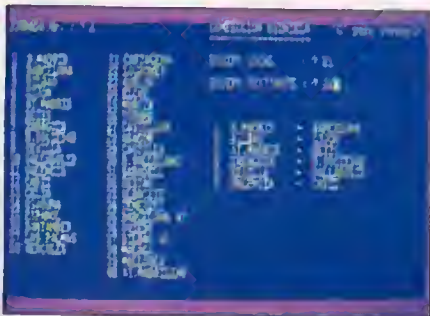
En el caso de una sola columna existen tres opciones. En la primera opción, una sola apuesta, se puede escoger el número de variantes. En la n.º 2, el usuario puede fijar para cada partido si será triple o doble. En la n.º 3, el usuario fija la cantidad de triples o dobles, asignando el ordenador los mismos, según un cálculo de probabilidades.

Para las boletos de ocho columnas, el usuario selecciona el número de fijos y el de variantes. Los fijos

pueden ser asignados por el usuario o por el ordenador.

El programa comienza con la actualización de la clasificación de los equipos de 1.ª y 2.ª división. Dicha clasificación sirve de base para el cálculo de probabilidades de cada partido de la quiniela. Como base del cálculo se ha establecido que con una diferencia de potencial «0», entre la clasificación del equipo local y la del visitante, existe un 50 por 100 de probabilidades de que el resultado sea «1», un 30 por 100 de «X» y un 20 por 100 de «2». Estas porcentajes varían en relación a las distintas diferencias de potencial de cada partido.

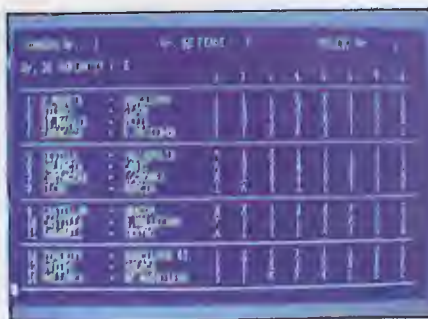
El programa puede ser considerado cuasialeatorio, debido al cálculo de probabilidades que efectúa para cada partido, teniendo en cuenta la «lógica» de la clasificación de los equipos.



Nota: el parpadeo de la pantalla indica que el programa está esperando a que se pulse una tecla.



Serie Oro



ESTRUCTURA

Líneas	Cometido
10-90	Inicio.
120-220	Menú inicial.
230-500	Actualización clasificación equipos de 1. ^a y 2. ^a
510-940	Confección de la quiniela.
950-1170	Menú de apuestas.
1180-1480	Resguardo quiniela de una columna. Las cifras entre paréntesis indican la clasificación de cada equipo.
1690-1770	Quiniela de una sola apuesta (1 columna).
1780-2000	Quiniela de una sola columna eligiendo el usuario los triples y los dobles. Calcula la diferencia de potencial de ambos equipos, la cual sirve de base para acotar el campo de posibilidades de cada signo. Un número aleatorio «sort» determina, según las probabilidades de cada signo, el resultado del partido.
2010-2350	Quiniela igual que la anterior, pero eligiendo el ordenador los triples y los dobles, según las distintas diferencias de potencial de los equipos.
2360-2720	Requisitos boletas de 8 columnas.
2730-2930	Resguardo quiniela boleta de 8 columnas.

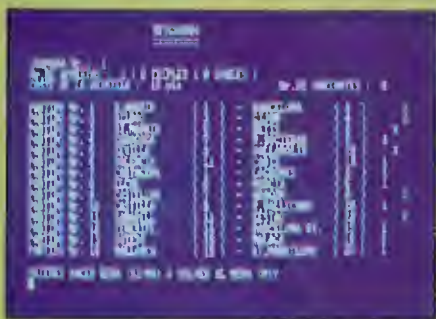
PRINCIPALES VARIABLES

eq&(x)	Equipo antes de clasificar.
eqcl&(x,y)	Equipo después de clasificar.
cl(x)	Variable usada para clasificar.
elq&(x)	Equipo local quiniela.
evq&(x)	Equipo visitante quiniela.
comb&(x)	Triple o doble.
celq(x)	Clasificación equipo local quiniela.
cevaq(x)	Clasificación equipo visitante quiniela.
uxd(x)	Indica el signo del resultado.
part(x)	Indica si es triple, doble o sencillo.
nt(x)	Número tabla diferencial de potencial.
nvar	Número de variantes.
mdos	Número máximo de doses.
dp	Diferencia de potencial.
pl	Probabilidades del signo «1».
pX	Probabilidades del signo «X».
sort	Número aleatorio para el cálculo de probabilidades.
nbol	Número de boletos.
nfij	Número de fijos.


```

10 '***** QUIN
IELA *****
20 'Autor: Josep Aracil i Xarrie
30 'Setiembre 1985
40 CLS:MODE 2:PAPER 0:PEN 1:clas1=0
:clas2=0
50 DIM eq$(20),eqcl$(2,20),cl(20),e
lq$(14),evq$(14),comb$(14),
celq(20),cevc(20),uxd(
14),part(14),nt(38)
60 SOUND 1,100,20,7
70 MODE 0:LOCATE 5,12:PRINT "1 - X
- 2"
80 GOSUB 100
90 MODE 2:GOTO 120
100 c$=INKEY$:IF c$="" THEN BORDER
1:BORDER 5:BORDER 1:BORDER 5:BORDER
1:
BORDER 7:GOTO 100
110 c$=UPPER$(c$):RETURN

```



```

120 '***** MENU INICIAL *****
****
130 CLS:SOUND 1,100,20,7
140 LOCATE 24,8:PRINT "MENU INICIAL
"
150 LOCATE 24,9:PRINT "=====
"
160 LOCATE 16,12:PRINT "1.- CONFEC
IONAR LA QUINIELA"
170 LOCATE 16,14:PRINT "2.- CARGAR
LA QUINIELA GRABADA"
180 LOCATE 7,18:PRINT "ELECCION ? :
"
190 GOSUB 100
200 e=VAL(c$):IF e<1 OR e>2 THEN 19
0
210 IF e=1 THEN GOTO 230
220 IF e=2 THEN GOSUB 3740:GOTO 950
230 '***** ACTUALIZACION CLASIFIC
ACIONES EQUIPOS *****
240 SOUND 1,100,20,7:div=1
250 IF div=1 THEN equips=18 ELSE eq
uips=20
260 CLS:PRINT "ACTUALIZACION CLASIF
ICACION ";div;" DIVISION"
270 PRINT "-----
":PRINT
280 FOR i=1 TO equips:cl(i)=0:NEXT
i
290 GOSUB 3470
300 FOR n=1 TO equips
310 LOCATE 1,n+3:PRINT eq$(n):LOCAT
E 20,n+3:PRINT "LUGAR EN LA CLASIFI
CACION : " ;:INPUT clas
320 IF clas>equips THEN 350
330 FOR i=1 TO n:IF cl(i)=clas THEN
350
340 NEXT i:GOTO 360
350 PRINT "ERROR EN EL NUMERO DEL E
QUIPO !!! ":LOCATE 30,n+3:PRINT cla
s:
GOSUB 100:PRINT "
":LOCA
TE 30,n+3:
PRINT "
":GOTO 310
360 cl(n)=clas:LOCATE 14,n+3:PRINT
clas

```

```

370 clas$=eq$(n):eqcl$(div,clas)=cl
as$
380 NEXT n:PRINT
390 CLS:PRINT "ACTUALIZACION CLASIF
ICACION ";div;" DIVISION"
400 PRINT "-----
":PRINT
410 FOR n=1 TO equips
420 PRINT n:TAB(5)eqcl$(div,n)
430 NEXT n
440 PRINT TAB(40)" ESTA BIEN LA CLA
SIFICACION (S/N) ?"
450 GOSUB 100
460 IF c$<>"S" AND c$<>"N" THEN 450
470 IF c$="N" THEN 250
480 IF c$="S" THEN IF div=1 THEN cl
as1=1:div=2:GOTO 250
490 IF c$="S" THEN IF div=2 THEN cl
as2=1:GOTO 510
500 GOTO 450
510 '***** CONFECION DE LA QUINI
ELA *****
520 CLS:SOUND 1,100,20,7
530 IF clas1=0 THEN PRINT "FALTA AC
TUALIZAR LA CLASIFICACION DE 1 DIVI
SION":
div=1:GOSUB 100:GOTO
250
540 IF clas2=0 THEN PRINT "FALTA AC
TUALIZAR LA CLASIFICACION DE 2 DIVI
SION":
div=2:GOSUB 100:GOTO
250
550 INPUT "JORNADA Nr. : ";jorn
560 div=1
570 LOCATE 1,4
580 FOR i=1 TO 18
590 PRINT i:TAB(5)eqcl$(div,i)
600 NEXT i
610 div=2
620 WINDOW #1,20,39,4,24
630 FOR j=1 TO 20
640 PRINT #1,j+20:TAB(5)eqcl$(div,j
)
650 NEXT j
660 WINDOW #2,40,80,1,25
670 PAPER #2,0:PEN #2,1
680 LOCATE #2,1,1:PRINT #2,"CONFEC
ION QUINIELA 'C' para corregir"
690 LOCATE #2,1,2:PRINT #2,"-----
"
700 FOR par=1 TO 14
710 LOCATE #2,19,4:PRINT #2,"
"
720 LOCATE #2,19,6:PRINT #2,"
"
730 LOCATE #2,1,8:PRINT #2,"
"
740 LOCATE #2,1,4:INPUT #2,"EQUIPO
LOCAL : ";x$
750 IF x$="c" OR x$="C" THEN par=pa
r-1:LOCATE #2,1,par+9:PRINT #2,"
"
:GOTO 700
760 LOCATE #2,1,6:INPUT #2,"EQUIPO
VISITANTE : ";y
770 x=VAL(x$)
780 IF x<19 AND y>19 THEN GOSUB 940
:GOTO 710
790 IF x>19 AND y<19 THEN GOSUB 940

```

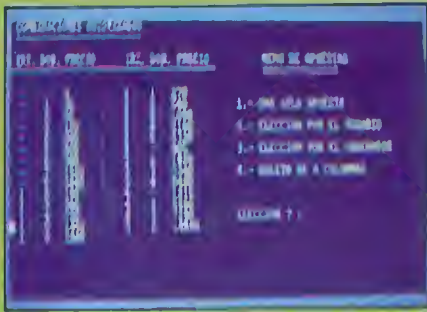
```

:GOTO 710
800 IF x=y THEN GOSUB 940:GOTO 710
810 IF x<19 THEN elq$(par)=eqcl$(1,
x):evq$(par)=eqcl$(1,y):
celq(par)=x:cevc(par)
=y
820 IF x>19 THEN elq$(par)=eqcl$(2,
x-20):evq$(par)=eqcl$(2,y-20):
celq(par)=x-20:cevc(p
ar)=y-20
830 FOR i=1 TO par-1:IF elq$(par)=e
lq$(i) OR elq$(par)=evq$(i) THEN 86
0
840 IF evq$(par)=evq$(1) OR evq$(pa
r)=elq$(i) THEN 860
850 NEXT i:GOTO 870
860 GOSUB 940:GOTO 710
870 LOCATE #2,1,par+9:PRINT #2,par;
TAB(5)elq$(par);TAB(17)"-";
TAB(20)evq$(par)
880 NEXT par
890 LOCATE #2,1,par+10:PRINT #2," E
STAS DE ACUERDO (S/N) "
900 GOSUB 100
910 IF c$<>"S" AND c$<>"N" THEN 900
920 IF c$="N" THEN 510
930 GOSUB 3550:GOTO 950
940 LOCATE #2,1,8:PRINT #2,"ERROR E
N EL Nr.DE LOS EQUIPOS":GOSUB 100:R
ETURN
950 '***** MENU DE APUESTAS *****
960 CLS:SOUND 1,100,20,7
970 PRINT " COMBINACIONES AUTORIZAD
AS"
980 PRINT " -----
--":PRINT
990 LOCATE 1,4:PRINT " TRI. DOB. P
RECIO TRI. DOB. PRECIO"
1000 LOCATE 2,6:PRINT " - -
0 2 1 270
- 1 30 2 2 540
- 2 60 2
3 1080"
1010 LOCATE 2,9:PRINT " - 3
120 2 4 2160
- 4 240 2 5 4320
- 5 480 3
- 405"
1020 LOCATE 2,12:PRINT " - 6
960 3 2 1620
- 7 1920 3 3 3240
- 8 3840 3
4 6480"
1030 LOCATE 2,15:PRINT " 1 2
180 4 - 1215
1 3 360 4 2 4860
1 4 720 5
1 7290"
1040 LOCATE 2,18:PRINT " 1 5

```

SUBROUTINAS

Líneas	Cometido
100-110	Subrutina «inkey».
1490-1680	Cálculo quiniela de 1 columna.
2940-3070	Cálculo boleto de 8 columnas.
3080-3240	Elección de fijos por el ordenador en relación a las diferencias de potencial.
3250-3350	Dibujo del boleto de 8 columnas.
3360-3460	Columna boleto de 8 columnas.
3470-3540	Clasificación inicial de los equipos de 1.ª y 2.ª
3550-3700	Grabación de la quiniela.
3740-3900	Lectura de la quiniela del cossette.



```

1440      5      2      14580
1      6      2880"
1050 PLOT 0,330:DRAWR 320,0
1060 PLOT 320,350:DRAWR 0,-270

1070 PLOT 150,350:DRAWR 0,-270:PLOT
320,350:DRAWR 0,-270
1080 LOCATE 50,4:PRINT "MENU DE APU
ESTAS"
1090 LOCATE 50,5:PRINT "=====
=="
1100 LOCATE 45,8:PRINT "1.- UNA SOL
A APUESTA"
1110 LOCATE 45,10:PRINT "2.- ELECCI
ON POR EL USUARIO"
1120 LOCATE 45,12:PRINT "3.- ELECCI
ON POR EL ORDENADOR"
1130 LOCATE 45,14:PRINT "4.- BOLETO
DE 8 COLUMNAS"
1140 LOCATE 45,18:PRINT "ELECCION ?
:"
1150 GOSUB 100
1160 e=VAL(c$):IF e<1 OR e>4 THEN 1
150
1170 ON e GOTO 1690,1780,2010,2360
1180 ***** RESGUARDO QUINIELA 1 C
OLUMNA *****
1190 GOSUB 1490
1200 IF dos<=mdos THEN 1210 ELSE 11
90
1210 IF op<>1 OR var$="A" THEN 1230
ELSE 1220
1220 IF va=nvar THEN 1230 ELSE 1190
1230 CLS:SOUND 1,100,20,7
1240 PRINT TAB(25)"RESGUARDO"
1250 PRINT TAB(25)"=====":PRINT
1260 PRINT "JORNADA Nr. : ";jorn
1270 ap=3*tri*2*dob:pre=ap*15
1280 PRINT "Nr. DE APUESTAS : "+STR$
(ap)+" ("+STR$(tri)+" TRIPLES "+ST
R$(dob)+" DOBLES )"
1290 PRINT "COSTE DE LA QUINIELA :
"+STR$(pre)+" pts";
1300 IF op=1 THEN PRINT TAB(49)"Nr.
DE VARIANTES : ";14-nvar
1310 FOR par=1 TO 14
1320 LOCATE 1,par+7:PRINT "PARTIDO
Nr. ";TAB(12)STR$(par);TAB(18)elq$(p
ar);
TAB(33)"(";TAB(34)ce
lq(par);TAB(38)";TAB(41)";TAB(4
4)evq$(par);
TAB(59)";
TAB(60)cevq(par);TAB(64)";"
1330 pro=uxd(par)
1340 ON pro GOTO 1350,1360,1370,138
0,1390,1400
1350 a$="1 " :GOTO 1410
1360 a$=" X " :GOTO 1410
1370 a$=" 2 " :GOTO 1410
1380 a$="1 X " :GOTO 1410
1390 a$=" X 2 " :GOTO 1410
1400 a$="1 X 2 "
1410 LOCATE 68,par+7:PRINT a$
1420 NEXT par
1430 PRINT :PRINT "QUIERES HACER OT
RA (S/N) O VOLVER AL MENU (M)?"
1440 GOSUB 100
1450 IF c$<>"S" AND c$<>"N" AND c$<
>"M" THEN 1440
1460 IF c$="M" THEN 950

```

```

1470 IF c$="S" THEN 1180 ELSE 1480
1480 END
1490 ***** CALCULO QUINIELA 1 COL
UMNA *****
1500 CLS:LOCATE 30,12:PRINT "CALCUL
ANDO . . ."
1510 va=0:dos=0
1520 FOR i=1 TO 14:uxd(i)=0:NEXT i
1530 FOR qui=1 TO 14
1540 dp=celq(qui)-cevq(qui)
1550 p1=50+dp*2
1560 px=30-dp
1570 sort=INT(100*RND+1)
1580 IF op=1 THEN 1640
1590 IF part(qui)=3 THEN uxd(qui)=6
:GOTO 1670
1600 IF part(qui)=2 THEN 1610 ELSE
1640
1610 IF sort>0 AND sort<=p1 TH
EN uxd(qui)=4:GOTO 1670
1620 IF sort>p1 AND sort<=p1+px TH
EN uxd(qui)=4:GOTO 1670
1630 IF sort>p1+px AND sort<=100 TH
EN uxd(qui)=5:GOTO 1670
1640 IF sort>0 AND sort<=p1 TH
EN uxd(qui)=1:va=va+1:GOTO 1670
1650 IF sort>p1 AND sort<=p1+px TH
EN uxd(qui)=2:GOTO 1670
1660 IF sort>p1+px AND sort<=100 TH
EN uxd(qui)=3:dos=dos+1:GOTO 1670
1670 NEXT qui
1680 RETURN
uxd1=1 / 2-
x / 3=2 / 4=1x / 5=x2 / 6=1x2
1690 ***** UNA SOLA APUESTA (1 CO
LUMNA) *****
1700 CLS:SOUND 1,100,20,7
1710 LOCATE 27,12:PRINT "CUANTAS VA
RIANTES QUIERES : "
1720 LOCATE 5,24:PRINT "TECLEA 'A'
SI QUIERES QUE SEA ALEATORIO"
1730 LOCATE 50,12:INPUT var$:nvar=V
AL(var$):nvar=14-nvar:var$=UPPER$(v
ar$)
1740 IF var$<>"A" THEN IF nvar<0 OR
nvar>14 THEN 1730
1750 LOCATE 22,14:INPUT "Nr. MAXIMO
DE 2 : ";mdos
1760 FOR i=1 TO 14:part(i)=0:NEXT
1770 op=1:GOTO 1180
1780 ***** ELECCION POR EL USUA
RIO (1 COLUMNA) *****
1790 CLS:SOUND 1,100,20,7:tri=0:dob
=0
1800 PRINT " ELECCION POR EL USUA
RIO"
1810 PRINT " -----
-"
1820 INPUT " Nr. DE TRIPLES : ";tri
1830 PRINT TAB(28)";CHR$(11);:INPU
T "Nr. DE DOBLES : ";dob
1840 PRINT TAB(53)";CHR$(11);:INPU
T "Nr. MAXIMO DE 2 : ";mdos:PRINT
1850 ap=3*tri*2*dob:pre=ap*15
1860 PRINT " Nr. DE APUESTAS : ";ap:
" PRECIO : ";pre;
" pts DE ACUERDO
? (S/N) "
1870 GOSUB 100:PRINT
1880 IF c$<>"S" AND c$<>"N" THEN 18
70
1890 IF c$="N" THEN 1780
1900 PRINT " QUINIELA JORNADA Nr. ";
jorn;TAB(30)"ELECCION COMBINACIONES
"
1910 PRINT " -----
-";TAB(30)"-----":
PRINT
1920 FOR i=1 TO 14:part(i)=0:NEXT
1930 FOR qui=1 TO 14
1940 PRINT qui;TAB(6)elq$(qui);"- "
;evq$(qui);TAB(30)
"(T riples-D obles-S
encillo) ";
1950 INPUT comb$(qui):comb$(qui)=UP
PER$(comb$(qui))
1960 IF comb$(qui)<>"T" AND comb$(q
ui)<>"D" AND comb$(qui)<>"S" THEN 1
950

```

Serie Oro

```

1970 IF comb$(qui)="T" THEN part(qu
i)=3:tri=tri+1
1980 IF comb$(qui)="D" THEN part(qu
i)=2:dob=dob+1
1990 NEXT qui:PRINT
2000 op=2:GOSUB 100:GOTO 1180
2010 ***** ELECCION POR EL ORD
ENADOR (1 COLUMNA) *****
2020 CLS:SOUND 1,100,20,7:tri=0:dob
=0:op=3
2030 LOCATE 20,6:PRINT "ELECCION PO
R EL ORDENADOR"
2040 LOCATE 20,7:PRINT "-----
-"
2050 LOCATE 20,10:INPUT "Nr. DE TRI
PLES : ";tri
2060 LOCATE 20,12:INPUT "Nr. DE DOB
LES : ";dob
2070 ap=3*tri*2*dob:pre=ap*15

2080 LOCATE 20,14:PRINT "Nr. DE APU
ESTAS : ";ap:" FRECIO : ";pr
e;" pts"
2090 LOCATE 20,16:INPUT "Nr. MAXIMO
DE 2 : ";mdos
2100 LOCATE 20,19:PRINT "ESTAS DE A
CUERDO ? (S/N)"
2110 GOSUB 100
2120 IF c$<>"S" AND c$<>"N" THEN 21
10
2130 IF c$="N" THEN 2010
2140 CLS:LOCATE 30,12:PRINT "CALCUL
ANDO . . ."
2150 RESTORE 2190
2160 FOR i=1 TO 38
2170 READ nt(i)
2180 NEXT i
2190 DATA -7,-6,-8,-5,-9,-4,-10,-3,
-11,-2
2200 DATA -12,-1,-13,-14,1,-15,2,-1
6,3,-17
2210 DATA 4,-18,5,-19,6,7,8,9,10,11
2220 DATA 12,13,14,15,16,17,18,19
2230 tr=0:do=0:FOR i=1 TO 14:part(i
)=0:NEXT
2240 FOR j=1 TO 38
2250 FOR qui=1 TO 14
2260 dp=celq(qui)-cevq(qui)
2270 IF part(qui)=3 THEN 2340
2280 IF part(qui)=2 THEN 2340
2290 IF tr=tri THEN IF dob=0 THEN 1
80
2300 IF dob>0 THEN IF do=dob THEN 1
80
2310 IF tr=tri THEN 2330
2320 IF dp=nt(j) THEN part(qui)=3:t
r=tr+1:GOTO 2340
2330 IF dp=nt(j) THEN part(qui)=2:d
o=do+1:GOTO 2340
2340 NEXT qui
2350 NEXT j
2360 ***** BOLETO DE 8 COLUMNAS
*****
2370 CLS:SOUND 1,100,20,7:op=4:f=0

2380 FOR i=1 TO 14:part(i)=0:NEXT
2390 LOCATE 20,7:PRINT "QUINIELA DE
8 COLUMNAS"
2400 LOCATE 20,8:PRINT "=====
=====
"
2410 LOCATE 20,11:INPUT "Nr. DE BOL
ETOS : ";nbol
2420 LOCATE 50,11:PRINT "PRECIO : "
;120*nbol;" pts"
2430 LOCATE 20,13:INPUT "Nr. DE FIO
DS (1) : ";finfj
2440 LOCATE 20,15:INPUT "Nr. DE VAR
IANTES : ";nvar:nvar=14-nvar

```


Serie Oro

```

2450 LOCATE 20,17:INPUT "Nr. MAXIMO
DE 2 : ";mdos:PRINT
2460 LOCATE 20,20:PRINT "ESTAS DE A
CUERDO (S/N) ?"
2470 GOSUB 100:PRINT
2480 IF c$<>"S" AND c$<>"N" THEN 24
70
2490 IF c$="N" THEN 2360
2500 CLS:LOCATE 30,10:PRINT " FIJOS
= 1 "
2510 LOCATE 30,11:PRINT " -----
"
2520 LOCATE 20,13:PRINT " LDS PONE
EL ORDENADOR (S/N) ? "
2530 GOSUB 100
2540 IF c$<>"S" AND c$<>"N" THEN 25
30
2550 IF c$="S" THEN 2730
2560 f=1:CLS:LOCATE 10,2:PRINT "QUI
NIELA"
2570 LOCATE 10,3:PRINT "-----"
2580 FOR qui=1 TO 14
2590 PRINT qui;TAB(7)elq$(qui);TAB(
20)";TAB(23)evq$(qui)
2600 NEXT qui
2610 LOCATE 40,2:PRINT "Nr. DE FIJO
S : ";nfij
2620 LOCATE 40,3:PRINT "-----
-----"
2630 FOR i=1 TO nfij
2640 LOCATE 52,9:PRINT " "
2650 LOCATE 40,7:PRINT "FIJO Nr. :
";i
2660 LOCATE 40,9:INPUT "PARTIT Nr.
: ";qui:part(qui)=1
2670 LOCATE 40,12+1:PRINT elq$(qui)
;TAB(54)";TAB(57)evq$(qui)
2680 NEXT i
2690 LOCATE 40,22:PRINT "ESTAS DE A
CUERDO (S/N) ?"
2700 GOSUB 100
2710 IF c$<>"S" AND c$<>"N" THEN 27
00
2720 IF c$="N" THEN 2500
2730 ***** RESGUARDO BOLETO DE 8
COLUMNAS *****
2740 CLS:SOUND 1,100,20,7
2750 LOCATE 22,12:PRINT "CALCULANDO
. . . . .":IF f=0 THEN GOSUB 3080
2760 FOR bol=1 TO nbol
2770 CLS:SOUND 1,100,20,7:PRINT
2780 PRINT " JORNADA Nr. : ";jorn;
2790 PRINT TAB(30)"Nr. DE FIJOS : "
;nfij;
2800 PRINT TAB(60)"BOLETO Nr. : ";b
ol
2810 PRINT:PRINT " Nr. DE VARIANT
ES : ";14-nvar
2820 GOSUB 3250
2830 FOR colum=1 TO 40 STEP 5
2840 GOSUB 2940:IF va=nvar AND dos<
=mdos THEN 2850 ELSE 2840
2850 GOSUB 3360
2860 NEXT colum
2870 GOSUB 100
2880 NEXT bol
2890 CLS:LOCATE 12,12:PRINT "QUIERE
S HACER OTRA (S/N) O VOLVER AL MENU
(M)?"
2900 GOSUB 100
2910 IF c$<>"S" AND c$<>"N" AND c$<

```

```

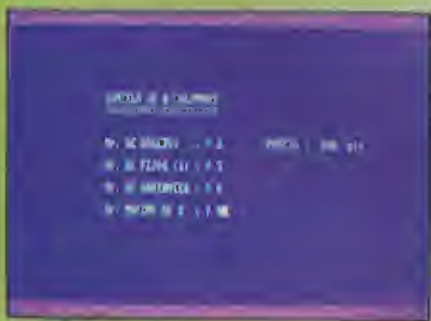
>"M" THEN 2900
2920 IF c$="M" THEN 950
2930 IF c$="N" THEN END ELSE 2360
2940 ***** CALCULO BOLETO 8 COLUMNA
S *****
2950 va=0:dos=0
2960 FOR i=1 TO 14:uxd(i)=0:NEXT i
2970 FOR qui=1 TO 14
2980 IF part(qui)=1 THEN uxd(qui)=1
:va=va+1:GOTO 3060
2990 dp=cclq(qui)-cevg(qui)
3000 p1=50+dp*2
3010 px=30-dp
3020 sort=INT(100*RND+1)
3030 IF sort>0 AND sort<=p1 TH
EN uxd(qui)=1:va=va+1:GOTO 3060
3040 IF sort>p1 AND sort<=p1+px TH
EN uxd(qui)=2:GOTO 3060
3050 IF sort>p1+px AND sort<=100 TH
EN uxd(qui)=3:dos=dos+1:GOTO 3060
3060 NEXT qui
3070 RETURN 'uxd1=1 / 2=
x / 3=2
3080 ***** ELECCION DE FIJOS POR EL
ORDENADOR *****
3090 RESTORE 3130
3100 FOR i=1 TO 19
3110 READ nt(i)
3120 NEXT i
3130 DATA 19,18,17,16,15,14,13,12,1
1,10
3140 DATA 9,8,7,6,5,4,3,2,1
3150 fijo=0:FOR i=1 TO 14:part(i)=0
:NEXT
3160 FOR j=1 TO 19
3170 FOR qui=1 TO 14
3180 dp=cclq(qui)-cevg(qui)
3190 IF fijo<nfij THEN 3240
3200 IF part(qui)=1 THEN 3220
3210 IF dp<nt(j) THEN part(qui)=1:f
ijo=fijo+1
3220 NEXT qui
3230 NEXT j
3240 RETURN
3250 ***** DIBUJO BOLETO 8 COLUMNA
S *****
3260 PLOT 15,312:DRAW 600,0
3270 PLOT 15,312:DRAW 0,-288:PLOT
295,312:DRAW 0,-288
PLOT 615,312:DRAW 0
,-288
3280 PLOT 15,232:DRAW 600,0:PLOT 1
5,152:DRAW 600,0
3290 PLOT 15,88:DRAW 600,0:PLOT 15
,24:DRAW 600,0
3300 LOCATE 40,5:PRINT "1 2 3
4 5 6 7 8"
3310 FOR n=1 TO 4:LOCATE 3,n+6:PRIN
T n;TAB(7)elq$(n);TAB(20)";TAB(23
)evq$(n): NEXT n
3320 FOR n=5 TO 8:LOCATE 3,n+7:PRIN
T n;TAB(7)elq$(n);TAB(20)";TAB(23
)evq$(n): NEXT n
3330 FOR n=9 TO 11:LOCATE 3,n+8:PRI
NT n;TAB(7)elq$(n);TAB(20)";TAB(2
3)evq$(n): NEXT n
3340 FOR n=12 TO 14:LOCATE 3,n+9:PR
INT n;TAB(7)elq$(n);TAB(20)";TAB(
23)evq$(n): NEXT n
3350 RETURN
3360 ***** COLUMNA BOLETO DE 8 CO
LUMNAS *****
3370 FOR n=1 TO 4:GOSUB 3420:LOCATE
colum+39,n+6:PRINT b$:NEXT n
3380 FOR n=5 TO 8:GOSUB 3420:LOCATE
colum+39,n+7:PRINT b$:NEXT n
3390 FOR n=9 TO 11:GOSUB 3420:LOCAT
E colum+39,n+8:PRINT b$:NEXT n
3400 FOR n=12 TO 14:GOSUB 3420:LOCA
TE colum+39,n+9:PRINT b$:NEXT n
3410 RETURN
3420 col=uxd(n)
3430 ON col GOTO 3440,3450,3460
3440 b$="1":RETURN
3450 b$="X":RETURN
3460 b$="2":RETURN
3470 ***** SUBROUTINA CLASIFICACIO
N INICIAL *****
3480 IF div=1 THEN RESTORE 3520:GOT

```

```

O 3500
3490 IF div=2 THEN RESTORE 3530
3500 FOR i=1 TO equips:READ eq$(i)
3510 NEXT i
3520 DATA BARCELONA,R.MADRID,BILBAO
,GIJON,AT.MADRID,BETIS,ZARAGOZA,R.S
OCIEDAD, SEVILLA,VALLADOLID,V
ALENCIA,CADIZ,CELTA,OSASUNA,ESPAÑOL
,SANTANDER, LAS PALMAS
,HERCULES
3530 DATA CARTAGENA,LOGRONES,HUELVA
,ELCHE,MURCIA,CORUNA,CASTELLON,ARAG
ON, CASTILLA,R.VALLECANO
,SABADELL,SESTAO,ALBACETE,TENERIFE.
BARCELONA AT., OVIEDO,BIL
BAD AT.,MALAGA,MALLORCA,AT.MADRILEN
O
3540 RETURN
3550 ***** GRABACION QUINIOLA *
*****
3560 CLS:SOUND 1,100,20,7
3570 LOCATE 29,10:PRINT "GRABACION
DE LA QUINIOLA"
3580 LOCATE 29,11:PRINT "-----
-----"
3590 LOCATE 26,13:PRINT "PARA DISCO
PULSAR CUALQUIER TECLA"
3600 LOCATE 12,15:PRINT "PARA CASSE
TTE PULSAR **REC y PLAY** DESPUES Q
UALQUIER TECLA"
3610 GOSUB 100
3620 SOUND 1,100,20,7
3630 CLS:LOCATE 27,12:PRINT "LA QUI
NIELA SE ESTA GRABANDO"
3640 OPENOUT "!quiniola"
3650 PRINT #9,jorn
3660 FOR qui=1 TO 14
3670 PRINT #9,elq$(qui)
3680 PRINT #9,evq$(qui)
3690 PRINT #9,cclq(qui)
3700 PRINT #9,cevg(qui)
3710 NEXT qui
3720 CLOSEOUT
3730 RETURN
3740 ***** LECTURA DEL CASSETTE
*****
3750 CLS:SOUND 1,100,20,7
3760 LOCATE 26,12:PRINT "PARA DISCO
PULSAR CUALQUIER TECLA"
3770 LOCATE 16,14:PRINT "PARA CASSE
TTE PULASR **PLAY**DESPUES QUALQUIE
R TECLA"
3780 GOSUB 100
3790 SOUND 1,100,20,7
3800 CLS:LOCATE 20,12:PRINT "LA QUI
NIELA SE ESTA CARGANDO EN MEMORIA"
3810 OPENIN "!quiniola"
3820 INPUT #9,jorn
3830 FOR qui=1 TO 14
3840 INPUT #9,elq$(qui)
3850 INPUT #9,evq$(qui)
3860 INPUT #9,cclq(qui)
3870 INPUT #9,cevg(qui)
3880 NEXT qui
3890 CLOSEIN
3900 RETURN

```



Para que tus dedo,
na realicen el trabajo duro, M.H. AMS
TRAD lo hace por ti. Todos los listados que incluyen
este logotipo se encuentran a tu disposición en un cas-
sette mensual, solicítalos.

YOUR COMPUTER

La Revista de ordenadores de mayor venta en toda Europa

¡SE PUBLICA DESDE AHORA EN ESPAÑA, EN FORMA DE CASSETTE!

Sí, ya está confirmada la sensacional noticia. Muy pronto estará **en los quioscos** de toda España una selección de los mejores juegos y utilidades publicados por la prestigiosa Revista británica «YOUR COMPUTER», editados en cassette de alta calidad y con instrucciones en castellano.

El **prestigio** alcanzado por Your Computer, tanto en Inglaterra como en España y otros países, se debe, de una forma muy especial, a la **gran**

calidad de los programas que publica, la mayor parte de ellos en Código Máquina, y con la utilización de rutinas y técnicas de programación muy depuradas.

Ahora, a un precio inmejorable, podéis tener acceso a estos programas, **evitandoos** la difícil tarea de **teclearlos** en vuestro ordenador.

¡Y **cada mes** estará en la calle una nueva cinta!

Si no encuentras la cassette de «Your Computer» en tu quiosco o tienda de informática, solicítala o nuestras oficinas:

SINTAX, S. A.

«YOUR COMPUTER»

Posea de la Castellana, 268

28046 Madrid

Envía tus señas completas, teléfono y **marca de ordenador** e incluye **talón bancario**, o remite **Giro Postal** por el importe.

No te cobraremos gastos por el envío.

Si prefieres pagar **contra reembolso**, entances incluye, junto a tu pedida, dos sellos de 50 ptas. cada uno para gastos de envío.

TAMBIEN DISPÓNIBLE
PARA

COMMODORE 64

y

SPECTRUM 48, PLUS, 128

1 YOUR COMPUTER

EL CORAZON DE LA PRIMERA REVISTA EUROPEA DE ORDENADORES

AMSTRAD

La mejor selección de programas de juegos y utilidades, publicados en la revista de mayor difusión de Europa en ordenadores.

Ahora reproducidos en cassette, en auténtica exclusiva mundial.

695.-
PTAS.

DYNAMITE DAN

El superagente Dan Dinamita penetra en la peligrosa y custodiada mansión del Dr. Blitzen, con la misión de abortar el proyecto del Rayo Devastador, de ello depende la seguridad del mundo libre.

SOLAMENTE
CPC 464

En el día 10-4-1921, los servicios de espionaje de Estados Unidos, descubrieron un plan de desarrollo de un arma revolucionaria, con un poder energético equivalente a 21.100 kg de nitroglicerina.

El Rayo Devastador, nombre dado a la poderosa arma por su creador, está en plena fase de desarrollo y sólo existe en el papel.

Su diseñador, el Dr. Blitzen una vez concluidos los estudios teóricos y terminados los 230 metros cuadrados de planos y las 1.100 páginas de instrucciones de montaje, ha vendido su siniestro artefacto al bloque del Este.

En su inespugnable mansión de los Alpes el Dr. y su equipo de seguridad, esperan la llegada del convoy que ha de transportar la documentación hasta sus compradores.

Transmitida la información a los servicios de inteligencia occidentales, éstos elaboran un plan para apoderarse de los documentos que pueden cambiar el destino del mundo.

En una audaz y arriesgada maniobra, los servicios de inteligencia consiguen comprar a la señora de la limpieza que presta sus servicios en la mansión.

Esta ha escondido cuidadosamente 8 cartuchos de dinamita, en diversos puntos de la casa, pero su afición por la bebida le ha hecho olvidar la situación de los mismos, cosa que no estaba prevista en los planes.

Una vez depositados los cartuchos entro en juego nuestro hombre; el superagente Dan dinamita.



Localizados éstos, el siguiente paso es llegar hasta la caja fuerte que guarda los planos del Rayo Devastador.

Encontrar el paredero de la caja y localizar los 8 cartuchos de dinamita, requiere eludir una larga serie de trampas y peligros, que sólo Dan puede conseguir, cualquier otro agente moriría en el empeño.

Después de todos estas peripecias volar la cerradura de la caja es una tarea de niños, con los planos en su poder, Dan puede decir que ha cubierto la mitad de su cometido.

Solamente le queda salir con vida de la mansión y entregar los planos al servicio de inteligencia.

Pero si penetrar en la mansión y descubrir el paradero de los planos era difícil, salir de ella cuesta mucho más, el cansancio, hace mella hasta en el agente secreto más famoso del mundo.

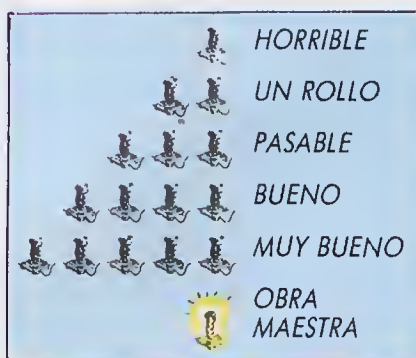


Depositado en el tejado de la misteriosa mansión por dirigible, comienza su arriesgada misión.

Pero la mansión no es una casa de campo normal y corriente, es una trampa mortal para cualquier extraño que quiera recorrerla. Pasadizos subterráneos, habitaciones letales, escaleras sin fin, pasillos subacuáticos, ascensores, trampas eléctricas y demás dispositivos extraños mantienen intacta la seguridad del malvado Blitzen.

Dan debe poner en constante riesgo su vida para salvar al mundo libre.

En primer lugar tiene que encontrar los ocho cartuchos de dinamita escondidos por la Sra. de la limpieza.





Mr. Joystick



Dan debe reponer fuerzas comiendo alimentos macrobióticos que están en determinados puntos de la casa: huevos, cócteles, frutas, queso, etc. ayudan a mantener la energía.

Otros objetos son imprescindibles para entrar en determinadas habitaciones y superar infernales corredores.

Dynamite Dan es el primer juego comercial en España que incorpora las siguientes opciones de juego:

Vidas infinitas.

Dan que aniquila bichas y objetos que le atacan.

Energía infinita.

Dan submarinista.

Opciones que se han añadido al juego normal, para facilitar el cumplimiento de la misión.

El juego en versión original era increíblemente difícil de resolver y en vista de esto se ha optado por añadir unas ayudas que permiten concluirlo con éxito.

Las distintas opciones de juego, pueden elegirse solas o combinadas, de forma que cada usuario pueda graduar a su gusto la dificultad de la misión.

Idea que permite resolver el juego utilizando todas las opciones primero, y luego una vez conocidos el emplazamiento de los objetos y la situación de los cartuchos de dinamita, ir poniendo el juego más difícil hasta conseguir completar la misión en la modalidad de juego normal.

Original idea en un juego comercial, que evita la desagradable experiencia de gastar dinero en un programa para entretenernos, y a las pocas horas tener que arrinconarlo porque nos resulta imposible pasar de un determinado punto, con el terrible pesar de haber tirado el dinero invertido.

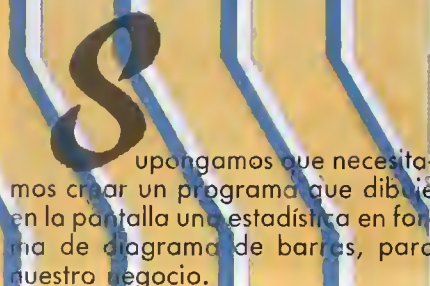
Esperamos que surjan más programas adaptados a este concepto.



ROTACION DE CARACTERES

Los soberbios gráficos de alta resolución del Amstrad son ideales para mostrar en pantalla gráficos y diagramas de todo tipo.

Haciendo uso de estos poderosos comandos gráficos, los datos pueden ser representados de una forma mucho más intuitiva y digerible para las personas.



Supongamos que necesitamos crear un programa que dibuje en la pantalla una estadística en forma de diagrama de barras, para nuestro negocio.

Normalmente, los ejes, escala y etiquetas de los ejes de este tipo de dibujos están pensados para ser observados verticalmente, mientras que el modo de impresión normal de una cadena de caracteres está implementada en sentido horizontal, de izquierda a derecha.

Si queremos que nuestras barras y texto aparezcan verticales, una forma de lograrlo sería la que usa el programa número 1.

No obstante, ésta no es una escritura vertical real; gire la cabeza 90 grados a derecha e izquierda y trate de leer el texto. Se entenderá perfectamente lo que quiero decir.

Parece ser que hace falta una rutina que se capaz de girar los caracteres de una cadena alfanumérica a derecha o izquierda 90 grados, e imprimirla en la pantalla en la posición del cursor de texto.

De nuevo el lenguaje máquina y las RSX

Naturalmente, hay que recurrir al lenguaje máquina para lograrlo, pero vamos a rizar un poco el rizo y, para nuestra mayor comodidad, crearemos dos nuevos comandos Basic que basta con llamarlos y darles la cadena para que lo hagan.

Como nuestros lectores ya saben, esto está «chupado» haciendo uso de las facilidades que el sistema operativo del Amstrad proporciona bajo el nombre de RSX, Extensiones Residentes del Sistema.

Uno de los nuevos comandos, «**PRINT.UP**», rota todos los caracteres de una cadena alfanumérica que se le proporciona como parámetro 90 grados en el sentido contrario a las agujas del reloj e imprime los caracteres verticalmente hacia arriba.

El otro, «**PRINT.DOWN**», hace lo mismo que su compañero, pero la impresión es de arriba a abajo y la rotación 90 grados en el sentido de las agujas del reloj.

El programa número 2 es una corta rutina en Basic que coloca el código máquina a partir de &A000 y activa los nuevos comandos, mientras que el programa 3 es el listado en lenguaje ensamblador de la rutina en máquina.

Los afortunados poseedores de un ensamblador podrán colocarlo en la zona de memoria que más les acomode y variar los nombres de los nuevos comandos si los propuestos por nosotros no les gustan.

Obsérvese que ambos deben ir precedidos, al invocarlos desde Basic, del signo «!» para que el sistema sepa a qué atenerse:

```
!PRINT.DOWN  
!PRINT.UP
```

y que el listado en ensamblador, aunque obtenido mediante el GENS de Hisoft, puede introducirse sin problemas usando el que publicamos en el número 8 de **AMSTRAD** Semanal.

Javier Igual



Sintaxis

La sintaxis de los nuevos comandos no puede ser más sencilla, basta con invocarlos y darles después la cadena e imprimir, bien entrecomillada o bien como variable:


```
frase$ = «Amstrad»  
!PRINT.UP, frase$  
!PRINT.DOWN, «Amstrad»
```

Aquellas personas que tengan un CPC464, deben usar solamente el método siguiente:

```
frase$ = «Amstrad»  
!PRINT.UP, @ frase$
```

La rutina en máquina, la que realmente hace el trabajo (el resto es la inicialización típica de un RSX), comienza en la dirección hexa &A031, inmediatamente debajo de la etiqueta «**AQUI**». Su funcionamiento es el siguiente:

```
a) LD L, (IX+0)  
LD H, (IX+1)
```

coloco en el registro HL la dirección de memoria donde se encuentra el descriptor de cadena que el sistema operativo construye (*recorrad que IX apuntaba a esa zona, denominada bloque de parámetros*).

b) LD B, (HL)

el primer byte del descriptor de cadena contiene su longitud, así que la cargamos en el registro B.

c) INC HL
LD E, (HL)
INC HL
LD D, (HL)
EX DE, HL

los dos bytes siguientes contienen la dirección verdadera de memoria donde los caracteres de la cadena se almacenan físicamente; la introducimos en el registro DE y luego en el HL, porque interesa que estén en él para el funcionamiento del resto de la rutina.

Alcanzamos ahora el bucle principal (*etiqueta «OTRO»*) que girará los caracteres y los imprimirá uno a uno mediante la instrucción DJNZ (de ahí que la longitud de la cadena esté en B).

Program Acción

A lo largo de este bucle hay que hacer unos cuantas cosas.

Lo primero de todo, es hollar la dirección donde está la matriz de puntos que compone el carácter que vamos a girar; esto lo hacemos llamando a la rutina firmware MATRIZ (*CALL MATRIZ*), que devuelve la dirección en HL.

A continuación activamos la ROM baja (*CALL ROMSI*), ponemos la dirección de la matriz de puntos en DE y llamamos a ARRIBA o ABAJO que giran el carácter o derecha o izquierda.

Deshabilitamos la ROM (*CALL ROMNO*) y convertimos la zona de memoria de nuestro programa que comienza en CHR255 en una matriz de puntos con el mismo formato que las nativas del **Amstrad** llamando a SETMAT (*CALL SETMAT*), de SET MATRIZ.


```

10 REM PROGRAMA 1
20 CLS
30 a$="AMSTRAD"
40 LOCATE 10,10
50 FOR i=1 TO LEN(a$)
60 PRINT MID$(a$,1,1);CHR$(10);CHR$(8);
70 NEXT

```

P

rogram Acción

<pre> A000 A000 A000 0109A0 A003 2124A0 A006 C3D1BC A009 11A0 A00B C328A0 A00E C32CA0 A011 5052494E A01A CE A01F 5052494F A022 D0 A023 00 A024 A02B 3E0A A02A 1802 A02C 3E0B A02E 3296A0 A031 DD6E00 A034 DD6601 A037 46 A03B 23 A039 5E A03A 23 A03B 56 A03C EB A03D 7E A03E E5 A03F C5 A040 CDA5BB A043 CD06B9 A046 EB A047 0E0B A049 3A96A0 A04C F5 A04D FE0A A04F CC74A0 A052 C485A0 A055 CD09B9 A05B 3EFF A05A 2197A0 A05D CDA8BB A060 3EFF A062 CD5ABB A065 F1 A066 CD5ABB </pre>	<pre> DIRORG: EQU #A000 ORG DIRORG LD BC,COMTAB LD HL,ESP_TR JP RSXCOM COMTAB: DEFW NDMTAB JP P_DOWN JP P_UP NDMTAB: DEFM "PRINT.DOW" DEFB "N"+#80 DEFM "PRINT.U" DEFB "F"+#80 DEFB #00 ESP_TR: DEFS 4 P_DOWN: LD A,#0A JR AQUI P_UP: LD A,#0B AQUI: LD (UPDOWN),A LD L,(IX+0) LD H,(IX+1) LD B,(HL) INC HL LD E,(HL) INC HL LD D,(HL) EX DE,HL LD A,(HL) DTRO: PUSH HL PUSH BC CALL MATRIZ CALL ROMSI EX DE,HL LD C,#0B LD A,(UPDOWN) PUSH AF CP #0A CALL Z,ARRIBA CALL NZ,ABAJO CALL ROMNO LD A,#FF LD HL,CHR255 CALL SETMAT LD A,#FF CALL PRINT POP AF CALL PRINT </pre>	<pre> A069 3E0B A06B CD5ABB A06E C1 A06F E1 A070 23 A071 10CA A073 C9 A074 2197A0 A077 1A A07B 060B A07A 17 A07E CB1E A07D 23 A07E 10FA A080 13 A081 00 A082 20F0 A084 C9 A085 219FA0 A08B 1A A089 060B A08B 2B A08C 17 A08D CB16 A08F 10FA A091 13 A092 00 A093 20F0 A095 C9 A096 00 A097 00 A098 00 A099 00 A09A 00 A09B 00 A09C 00 A09D 00 A09E 00 A09F 00 B906 B909 BB5A BBA5 BBAB BCD1 A0A0 </pre>	<pre> LD A,#0B CALL PRINT POP BC POP HL INC HL DJNZ DTRO RET ARRIBA: LD HI,CHR255 LD A,(DE) LD B,#0B BUCLE1: RLA RR (HL) INC HL DJNZ BUCLE1 INC DE DEC C JR NZ,ARRIBA RET ABAJO: LD HL,11_CHAR LD A,(DE) LD B,#0B BUCLE2: DEC HL RLA RL (HL) DJNZ BUCLE2 INC DE DEC C JR NZ,ABAJO RET UPDOWN: DEFB 0 CHR255: DEFB 0 DEFB 0 DEFB 0 DEFB 0 DEFB 0 DEFB 0 DEFB 0 DEFB 0 DEFB 0 DEFB 0 N_CHAR: DEFB 0 ROMSI: EQU #B906 ROMNO: EQU #B909 PRINT: EQU #BB5A MATRIZ: EQU #BBAB SETMAT: EQU #BBAB RSXCOM: EQU #BCD1 END </pre>
---	---	---	---

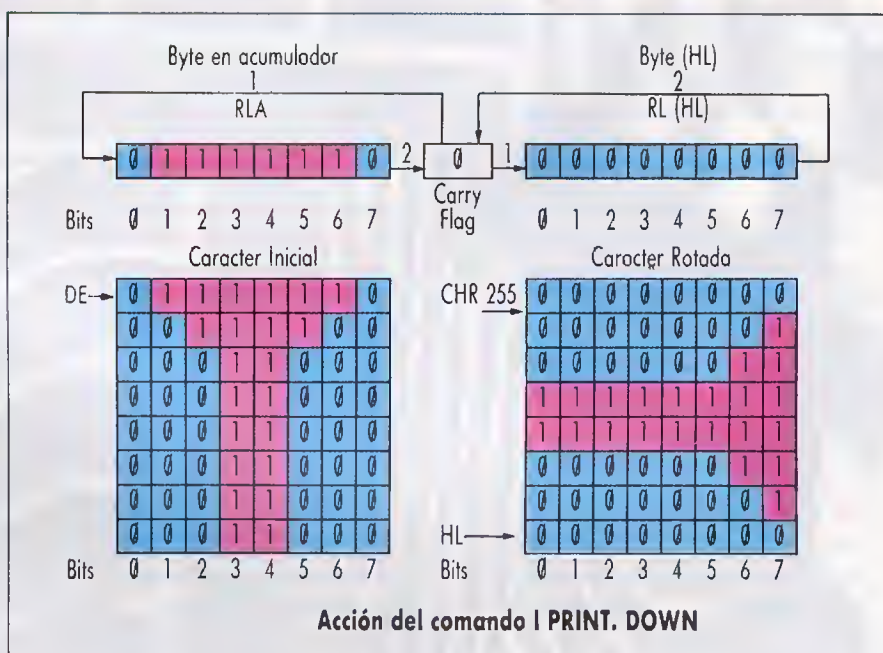


Figura 1

Por fin imprimimos el caracter y repetimos el ciclo.

Rutinas de giro

Vamos ahora con las rutinas ARRIBA y ABAJO.

Si se mira con detalle la rutina ARRIBA, se puede ver que consta de dos bucles.

El primero recorre los 8 bytes que forman un caracter de uno en uno, cargando el A con el contenido del registro DE, que es donde la rutina MATRIZ, tras el EX DE, HL, deja la dirección requerida.

El segundo bucle, más interno, rota los caracteres bit a bit: se rota el acumulador hacia la izquierda, de forma que el bit 7 acaba en el Carry flag (RLA).

Luego, este bit se incluye en el

apropiado byte de CHR255 con la instrucción de rotación a la derecha RR (HL).

El resultado de estas manipulaciones es que la trama de bytes y bits de CHR255 contiene al final una copia exacta del caracter original que estaba en el acumulador, pero girado 90 grados.

Ya habréis podido observar que todo esto resulta extremadamente sencillo de manejar gracias al sistema RSX y a las facilidades previstas por el firmware del **Amstrad**. Los dos nuevos comandos ayudarán a que vuestros gráficos estén dotados de una presentación mucho más agradable y profesional.

Sólo para divertirse ¿alguien podría adaptar la rutina para que el texto resultante apareciera tal y como se vería en un espejo?

LLAMADAS AL FIRMWARE

ETIQUETA	DIRECCION	PROPOSITO
ROMSI	&B906	Activo lo ROM bajo donde se encuentra el firmware (direcciones &0-&3FFF).
ROMNO	&B909	Desactivo la ROM bajo.
PRINT	&BB5A	Imprime el caracter contenido en el acumulador.
MATRIZ	&BBA5	Encuentra la dirección de la matriz que constituye el corocter dado como parámetro en el acumulador. Devuelve dicho dirección en el registro HL.
SETMAT	&BBA8	Convierte uno zona de memoria de 8 bytes al formato necesario para que represente un caracter alfanumérico. Requiere dos parámetros: el código ASCII del carocter en el acumulador y la dirección de lo zona de memoria en HL.
RSXCOM	&BCD1	Le dice al firmware que existe una RSX. Necesito dos parámetros: en BC, lo dirección de la tabla de nombres de la RSX, y en HL la dirección de un espacio de trabajo (ESP_TR) de 4 bytes de longitud.

Program Acción

ETIQUETAS

```

ABAJO  A0B5  A0U1  A02E  ARRIBA  A074
BUCLE1  A07A  BUCLE2  A0B8  CHR255  A097
COMTAB  A009  DIRORG  A000  ESP_TR  A024
MATRIZ  BBA5  NOMTAB  A011  N_CHAR  A09F
OTRO    A03D  PRINT  BBA5  P_DOWN  A02B
P_UP    A02C  ROMNO  B909  ROMSI  B906
RSXCOM  BCD1  SETMAT  BBA8  UPDOWN  A096

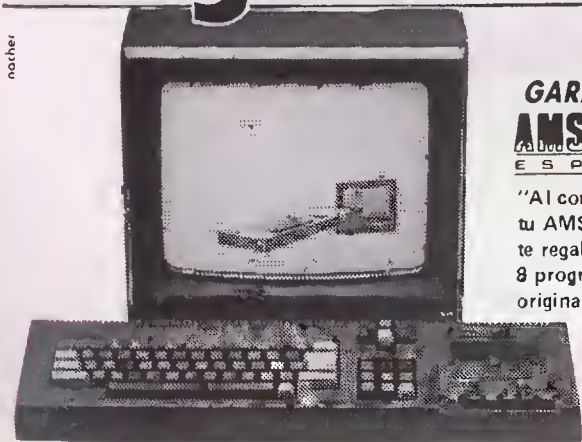
```

```

10 REM Side Writer
20 REM By R.A.Waddilove
30 REM (c)Amstrad Semanal
40 MEMORY &9FFF
50 a=&A000
60 FOR i=1 TO 14
70 sum=0:READ a$,b$
80 FOR j=1 TO 21 STEP 2
90 k=VAL("&" + MID$(a$,j,2)):POKE a,i
100 sum=sum+k:a=a+1
110 NEXT
120 IF sum<>VAL("&" + b$) THEN PRINT
"Error in data...":STOP
130 NEXT
140 CALL &A000
150 PRINT "Side writer esta listo..
."
160 PRINT "Dispones de los comandos
'i:print.up' y 'i:print.down'"
170 DATA 0109A02124A0C301BC11A0,490
180 DATA C32BA0C32CA05052494E54,4A7
190 DATA 2E444F57CE5052494E542E,3A1
200 DATA 55D0000000000000E0A1B02,1B7
210 DATA 3E0B3296A0DD6E00DD6601,440
220 DATA 46235E2356EB7EE5C5CDA5,5C5
230 DATA BB0D06B9EB0E0B3A96A0F5,5AD
240 DATA FE0ACC74A0C4B5A0CD09B9,660
250 DATA 3EFF2197A0CDABBB3EFFCD,6CF
260 DATA 5ABBF1CD5ARB3E0BCD5ABB,610
270 DATA C1E12310CAC92197A01A06,4E0
280 DATA 0B17CB1E2310FA130D20F0,365
290 DATA C9219FA01A060B2B17CB16,374
300 DATA 10FA130D20F0C900000000,303

```

Sigue la línea del futuro



**GARANTIA
AMSTRAD
ESPAÑA**

"Al comprar
tu AMSTRAD
te regalamos
8 programas
originales"

ORDENADOR AMSTRAD 464 F. Verde 66.900 Ptas.
ORDENADOR AMSTRAD 6.128 F. Verde 109.500 Ptas.

DISTRIBUIDOR INDESCOMP PARA ORENSE:



Almacenes Méndez

CAPITAN CORTES, 17. TELEF.-228607. ORENSE

Servimos a tiendas y almacenes

Presenta este anuncio y obtendrás un OBSEQUIO en tu compra

ROBOT RON

Las pulgas mutantes atacan de nuevo, como cada primavera, año tras año.

No hay problema. ¡Enviad a buscar a Robot Ron!

Ron es un robot dirigido por control remoto especializado en destruir pulgas mutantes y armado de un super láser, por ejemplo. Pulga que pilla, pulga que mata.

Como era de esperar, nuestro robot no es inmune al ataque de las ominosas pulgas, en cuanto le tocan a él, lo destruyen.

Bueno, no os preocupéis. Podemos reconstruirlo, para eso está la tecnología, pero sólo tres veces. **Algún límite tiene que haber, ¿no?**

El programa consta de nueve pantallas distintas donde puedes probar tu habilidad, velocísimos reflejos, etc.

A medida que vamos avanzando por las pantallas, hay menos pulgas, pero se mueven con mayor agilidad y mala idea.

Hemos usado una rutina en máquina para pintar las pulgas, el robot y el disparo, la cual cumple su función unas 10 veces más rápido que el básico.

Ojo con las Datas, son muchas y es fácil equivocarse.



TABLA DE VARIABLES

x,y	Coordenadas de Ron.
a(20,25)	Mapa de pantalla.
r(20,1)	Coordenadas de las pulgas.
name\$(5)	Nombres en la tabla de puntos.
hiscore(5)	Máxima puntuación
p,q	Coordenadas del disparo.
sc	Puntos.
screen	Número de pantalla.
m	Desplazamiento a la izquierda de las pulgas.
n	Pulga que será movida.

```

10 REM Robot Ron v The Evil Weevils
20 REM By R.A.Waddilove
30 REM(c) Amstrad Semanal
40 MEMORY &9FFF
50 GOSUB 420:REM inicializacion
60 GOSUB 1530:REM instrucciones
70 WHILE NOT fed.up
80 GOSUB 1230:REM scores
90 WHILE lives
100 GOSUB 670:REM asigna variables
110 WHILE a(x,y)=0 AND m
120 GOSUB 200:GOSUB 260:IF p THEN G
OSUB 390 ELSE GOSUB 360
130 WEND
140 IF m THEN GOSUB 930 ELSE screen
=screen+(screen>1):GOSUB 910
150 WEND
160 IF sc>hiscore(5) THEN GOSUB 135
0
170 WEND
180 END
190 REM -----mueva hombre -----
----
200 i=x+(INKEY(g)>1)-(INKEY(h)>1)
:j=y+(INKEY(e)>1)-(INKEY(f)>1):IF
a(i,j)>1 THEN RETURN
210 IF a(i,j)>1 THEN RETURN
220 CALL &A000,x,y,i,j,2:IF p=0 AND
(i<>x OR j<>y) THEN mx=i-x:my=j-y:
SOUND 132,100,1,4
230 x=i:y=j
240 RETURN
250 REM ----- mueve robots -----
---
260 SOUND 130,m*20+500,10,4:i=r(n,0
):j=r(n,1):IF i=0 THEN 300
270 IF a(i,j)<>1 THEN LOCATE i,j:PR
INT " ":r(n,0)=0:a(i,j)=0:m=m-1:GOT
O 300
280 i=i-(i<x)+(i>x):j=j+(j>y)-(j<y)
290 IF a(i,j)=0 THEN CALL &A000,r(n
,0),r(n,1),i,j,0:a(r(n,0),r(n,1))=0
:a(i,j)=1:r(n,0)=1:r(n,1)=j
300 i=r(n+1,0):j=r(n+1,1):IF i=0 TH
EN 340
310 IF a(i,j)<>1 THEN LOCATE i,j:PR
INT " ":r(n+1,0)=0:a(i,j)=0:m=m-1:G
OTO 340
320 i=i-(i<x)+(i>x):j=j+(j>y)-(j<y)
330 IF a(i,j)=0 THEN CALL &A000,r(n
+1,0),r(n+1,1),i,j,1:a(r(n+1,0),r(n
+1,1))=0:a(i,j)=1:r(n+1,0)=1:r(n+1,
1)=j
340 n=(n+2)MOD screen*2:IF r(n,0) 0
R r(n+1,0) OR m=0 THEN RETURN ELSE
340
350 REM ----- fuego -----
360 IF INKEY(d)<0 THEN RETURN
370 p=x+mx:q=y+my:IF a(p,q)=3 THEN
p=0:RETURN ELSE IF a(p,q)=1 THEN CA
LL &A000,p,q,p,q,4:p=0:a(p,q)=4:RFT
URN
380 CALL &A000,p,q,p,q,3:a(p,q)=7:S
OUND 129,10,300,15,1,1,1:RETURN
390 i=p+mx:j=q+my:IF a(i,j)=1 THEN
a(p,q)=0:a(i,j)=4:CALL &A000,p,q,i,
j,4:p=0:SOUND 129,1500,100,15,2,0,1
5:sc=sc+1:LOCATE 7,1:PRINT MID$(STR
$(sc),2):RETURN
400 a(p,q)=0:IF a(i,j)=3 THEN LOCAT
E p,q:PRINT " ":p=0:SOUND 129,0,0,0
ELSE CALL &A000,p,q,i,j,3:a(i,j)=2
:p=1:q=j
410 RETURN
420 REM ----- inicializar -----
---
430 DEFINT a-z
440 DIM a(20,25),r(20,1),name$(5),h
iscore(5)
450 FOR i=1 TO 5:name$(i)="-F" grupo
A":hiscore(i)=60-i*10:NEXT
460 ENT 1,200,2,1:FNV 1,15,-1,25
470 FNV 2,15,-1,10
480 FOR i=0 TO 98
490 READ a$:POINT &A000+i,VAL("&"+a$
)
500 NEXT

```



```

510 DATA DD,7F,00,87,87,87,87,7E
,47,80,26,00,DD,7F,04,3D,87,87,6F,1
1,50,00,DD,46,02,05,19,10,FD,FF,26,
00,DD,7F,08,3D,87,87,6F,11,50,00,DD
,46,06,05,19,10,FD,11,FD,07,06,03,7
6,00,27,76,00,23,76,00,27,76,00
520 DATA 19,10,FF,FF,11,00,A1,06,08
,1A,77,13,27,1A,77,13,27,1A,77,13,2
3,1A,77,17,78,01,FD,07,19,47,10,FF,
C9
530 FOR i=0 TO 19:READ J:POKE &A10
0+i,J:NEXT
540 REM Robot 1
550 DATA 20,60,60,0,105,131,195,40,
105,169,233,40,41,7,3,40,20,0,20,0,
0,40,40,0,0,160,160,0,80,160,240,0
560 REM Robot 2
570 DATA 80,240,240,160,160,1,3,80,
160,169,86,80,161,3,3,82,80,2,1,160
,0,240,240,0,20,0,0,40,40,0,0,20
580 REM Hombre
590 DATA 0,64,128,0,0,128,64,0,0,64
,128,0,4,12,12,8,8,76,140,4,72,12,1
2,16,0,8,4,0,4,8,4,8
600 REM Laser
610 DATA 0,0,0,0,0,8,4,0,0,0,0,0,0,
84,168,0,0,84,168,0,0,0,0,0,0,8,4,0
,0,0,0,0
620 REM Explosion
630 DATA 8,84,168,4,4,176,68,8,168,
8,4,84,84,216,228,168,84,216,228,16
8,168,8,4,84,4,136,68,8,8,84,168,4
640 FOR i=1 TO 20:a(i,2)=3:a(i,24)=
3:NEXT
650 FOR i=2 TO 24:a(i,1)=3:a(20,i)=
3:NEXT
660 RETURN
670 REM -----a=igna variables -
680 CLS #1:PEN 1
690 FOR i=2 TO 19:FOR j=3 TO 23:a(i
,j)=0:NEXT:NEXT
700 j=1+INT(RND*5):DEG
710 FOR i=0 TO 350 STEP 10
720 x=10+6*COS(i*J):y=12+6*SIN(i):L
OCATE x,y:PRINT CHR$(207):a(x,y)=5
730 NEXT
740 x=10:y=12:p=0:a(x,y)=0
750 m=screen*2
760 FOR i=0 TO m-1
770 IF RND>0.5 THEN r(i,0)=2+INT(RN
D*5) ELSE r(i,0)=15+INT(RND*5)
780 IF RND>0.5 THEN r(i,1)=3+INT(RN
D*5) ELSE r(i,1)=15+INT(RND*5)
790 IF a(r(i,0),r(i,1)) THEN 770 EL
SE a(r(i,0),r(i,1))=1
800 NEXT
810 REM ----- crea pantalla -----

```

```

820 MOVE 0,18:DRAW 0,380,1:DRAW 638
,380:DRAW 638,18:DRAW 0,18
830 PEN 15:LOCATE 1,1:PRINT "Score:
" MID$(STR$(sc),2):PEN 14:PRINT TAB
(12)"High:" MID$(STR$(hiscor(1)),2)
840 PEN 13:LOCATE 1,25:PRINT "Lives
:" MID$(STR$(lives),2):PEN 12:PRINT
TAB(12)"Screen:" MID$(STR$(10-scre
e),2)
850 CALL &A000,x,y,x,y,2
860 PEN 4
870 FOR i=0 TO screen*2-1
880 LOCATE r(i,0),r(i,1):PRINT CHR$
(231)
890 NEXT
900 PEN 15
910 FOR i=0 TO 2000:NEXT
920 RETURN
930 REM ----- muerte -----
-
940 SOUND 129,1500,100,15,2,0,15
950 lives=lives-1
960 CALL &A000,x,y,x,y,4
970 FOR i=0 TO 5000:NEXT
980 IF lives THEN RETURN
990 TAB
1000 PLOT -10,-10,2:MOVE 160,200:PR
INT "Game Over";
1010 TAGOFF:PRINT CHR$(23);CHR$(1);
:TAG
1020 PLOT -10,-10,6:MOVE 164,202:PR
INT "Game Over";
1030 TAGOFF:PRINT CHR$(23);CHR$(0);
1040 FOR i=0 TO 5000:NEXT
1050 RETURN
1060 REM ----- sonido -----
1070 WHILE INKEY(47)<0 AND INKEY(76
)<0
1080 RESTORE 1190:READ pitch:SOUND
129,213,200,4:SOUND 130,253,2000,4:
SOUND 1,127,2000,4
1090 WHILE pitch>0 AND INKEY(47)<0
AND INKEY(76)<0
1100 SOUND 4,pitch,10,6:SOUND 4,159
,10,6:READ pitch
1110 WEND
1120 READ pitch:SOUND 129,319,200,5
:SOUND 1,213,200,4:SOUND 1,159,200,
4
1130 WHILE pitch>0 AND INKEY(47)<0
AND INKEY(76)<0
1140 SOUND 4,pitch,12,6:READ pitch
1150 WEND
1160 WEND
1170 IF INKEY(47)>-1 THEN d=47:e=69
:f=71:g=39:h=31 ELSE d=76:e=72:f=73
:g=74:h=75

```

Serie Oro



```

1180 RETURN
1190 DATA 142,127,119, 142,127,119,
127,119,106, 127,119,106, 119,106,
95, 119,106,95, 106,95,84, 106,95,8
4,0
1200 DATA 80,84,95, 84,95,106, 95,1
06,119, 106,119,127
1210 DATA 80,84,95, 84,95,106, 95,1
06,119, 106,119,127
1220 DATA 119,127,142,159,169,190,2
13,239,253,284,319,253,284,319,239,
253,284,213,239,253,190,213,239,169
,190,213,190,179,169,159,142,159,0
1230 REM ----- scores -----
1240 MODE 1:BORDER 1
1250 DRAW 0,398,1:DRAW 638,398:DRAW
638,0:DRAW 0,0
1260 PAPER 2:PEN 3:LOCATE 11,2:PRIN
T " High Scores ":PAPER 0
1270 FOR i=1 TO 5
1280 LOCATE 7,5+i*2:PEN 3:PRINT i:
PEN 1:PRINT " :name$(i):" ":PEN 2
:WHILE POS(#0)<29:PRINT ".":WEND:P
EN 3:PRINT hiscor(i)
1290 NEXT
1300 PAPER 1:PEN 0:LOCATE 5,24:PRIN
T " Pulsa SPACE o FIRE ":PAPER 0
1310 WHILE INKEY<>"":WEND:GOSUB 10
60

```

MICROHOBBY AMSTRAD SEMANAL

LE OFRECE AHORA SUS PROGRAMAS YA GRABADOS, PARA QUE VD. NO TENGA QUE TECLEARLOS

Todos los programadores y aficionados a la microinformática sabemos lo tedioso y propenso a errores que resulta el teclear un listado de un programa. Para facilitar tu labor al máximo y que no tengas que estar horas sobre el teclado de tu ordenador tratando de descifrar incomprensible mensajes de error, **AMSTRAD SEMANAL** te ofrece cada mes los programas publicados de los cuatro números correspondientes en una cinta de cassette, sólo por **675 ptas. (sin más gastos por envío).**

Envíanos con la menor demora posible, el cupón correspondiente.


```

1320 SOUND 129,0:SOUND 130,0:SOUND
132,0
1330 BORDER 0:MODE 0:WINDOW #1,2,19
,3,23:sc=0:screen=9:lives=3
1340 RETURN
1350 REM ----- high score -----
1360 MODE 1
1370 DRAW 0,398,1:DRAW 638,398:DRAW
638,0:DRAW 0,0
1380 PEN 1:PAPER 0:LOCATE 7,5:PRINT
"Nueva puntuacion maxima"
1390 LOCATE 10,10:PEN 3:PRINT "Dime
tu nombre"
1400 LOCATE 14,20:PAPER 2:PEN 3:PRI
NT " 2..... "
1410 LOCATE 16,20:k$="":name$(5)=""
1420 WHILE INKEY$<>"":WEND
1430 WHILE k$=CHR$(13)
1440 IF k$=CHR$(71) AND k$<"z" AND
LEN(name$(5)) < 10 THEN name$(5)=name
$(5)+k$:PRINT k$
1450 IF k$=CHR$(127) AND LEN(name$(
5)) THEN name$(5)=LEFT$(name$(5),LE
N(name$(5))-1):PRINT CHR$(8);CHR$(1
6):
1460 k$=INKEY$
1470 WEND
1480 hiscore(5)=sc
1490 FOR i=5 TO 2 STEP -1
1500 IF hiscore(i)>hiscore(i-1) THE
N k$=name$(i):name$(i)=name$(i-1):n
ame$(i-1)=k$:sc=hiscore(i):hiscore(
i)=hiscore(i-1):hiscore(i-1)=sc
1510 NEXT
1520 RETURN
1530 REM ----- instrucciones -----
1540 MODE 1:CALL &BB4E:CALL &BC02
1550 INK 2,2:INK 0,11:INK 3,11:BORD
ER 11
1560 WINDOW #1,11,31,1,3:PAPER #1,3
:CLS #1
1570 WINDOW #1,10,30,2,4:PAPER #1,1
:CLS #1
1580 PEN 3:LOCATE 1,25:PRINT "Robot
Ron":PLOT -10,-10,2
1590 FOR x=0 TO 158 STEP 2:FOR y=2
TO 14 STEP 2
1600 IF TEST(x,y) THEN PLOT x+230,y
+y+344:PLOT x+230,y+y+346
1610 NEXT:NEXT:PEN #1,2

```

```

1620 WINDOW #1,2,40,5,24:PAPER #1,3
:CLS #1:INK 3,6
1630 WINDOW #1,1,39,6,25:PAPER #1,1
:CLS #1
1640 PRINT #1
1650 PRINT #1,"Invasion de pulgas g
igantes!."
1660 PRINT #1
1670 PRINT #1," Estas criaturas deb
en ser exterminadas"
1680 PRINT #1," a cualquier precio.
":PRINT #1
1690 PRINT #1," Enviad a Robot Ron
para que pueda"
1700 PRINT #1," destruirlas con su
super laser."
1710 PRINT #1
1720 PRINT #1," Ron puede controlar
se usando las"
1730 PRINT #1," siguientes teclas..
.
1740 PRINT #1
1750 PRINT #1," A=ar Z=abaj
<=izq >=der"
1760 PRINT #1
1770 PRINT #1,SPC(11)"SPACE dispara
"
1780 PRINT #1
1790 PRINT #1,SPC(11)"O use Joystic
k"
1800 PEN #1,1:PAPER #1,2:LOCATE #1,
3,19:PRINT #1," Pulsa SPACE o FIRE
para comenar ":PAPER #1,0
1810 WHILE INKEY$<>"":WEND
1820 GOSUB 1060
1830 RESTORE 1840:FOR x=0 TO 15:REA
D j:INK 1,j:NEXT:INK 7,26,13
1840 DATA 0,18,6,24,2,8,20,26,15,7,
7,9,25,16,22,23
1850 RETURN

```

```

A000 DD7E00 LD A,(IX+0)
A003 87 ADD A,A
A004 87 ADD A,A
A005 87 ADD A,A
A006 87 ADD A,A

```

Serie Oro

```

A007 87 ADD A,A
A008 3247A0 LD (LA047),A
A00B 26C0 LD H,#C0
A00D DD7E04 LD A,(IX+4)
A010 7D DFC A
A011 87 ADD A,A
A012 87 ADD A,A
A013 6F LD I,A
A014 115000 LD DE,#0050
A017 DD4602 LD B,(IX+2)
A01A 05 DFC B
A01B 19 LA01R ADD HL,DE
A01C 10FD DJNZ LA01B
A01E F5 PUSH HL
A01F 26C0 LD H,#C0
A021 DD7E08 LD A,(IX+8)
A024 7D DFC A
A025 87 ADD A,A
A026 87 ADD A,A
A027 6F LD I,A
A028 115000 LD DE,#0050
A02B DD4606 LD B,(IX+6)
A02E 05 DFC I
A02F 19 LA02F ADD HL,DE
A030 10FD DJNZ LA02F
A032 11FD07 LD DE,#07FD
A035 0608 LD R,#08
A037 3600 LA037 LD (HL),#00
A039 23 INC HL
A03A 3600 LD (HL),#00
A03C 23 INC HL
A03D 3600 LD (HL),#00
A03F 23 INC HL
A040 3600 LD (HL),#00
A042 19 ADD HL,DE
A043 10F2 DJNZ LA037
A045 E1 POP HL
A046 1100A1 LD DE,#A100
A049 0608 LD R,#08
A04B 1A LA04B LD A,(DE)
A04C 77 LD (HL),A
A04D 13 INC DE
A04E 23 INC HL
A04F 1A LD A,(DE)
A050 77 LD (HL),A
A051 13 INC DE
A052 23 INC HL
A053 1A LD A,(DE)
A054 77 LD (HL),A
A055 13 INC DE
A056 23 INC HL
A057 1A LD A,(DE)
A058 77 LD (HL),A
A059 13 INC DE
A05A 7B LD A,R
A05B 01FD07 LD BC,#07FD
A05E 09 ADD HL,BC
A05F 47 LD B,A
A060 10E9 DJNZ LA04B
A062 C9 RET

```



Pon

DINAMITA

a tu imaginación

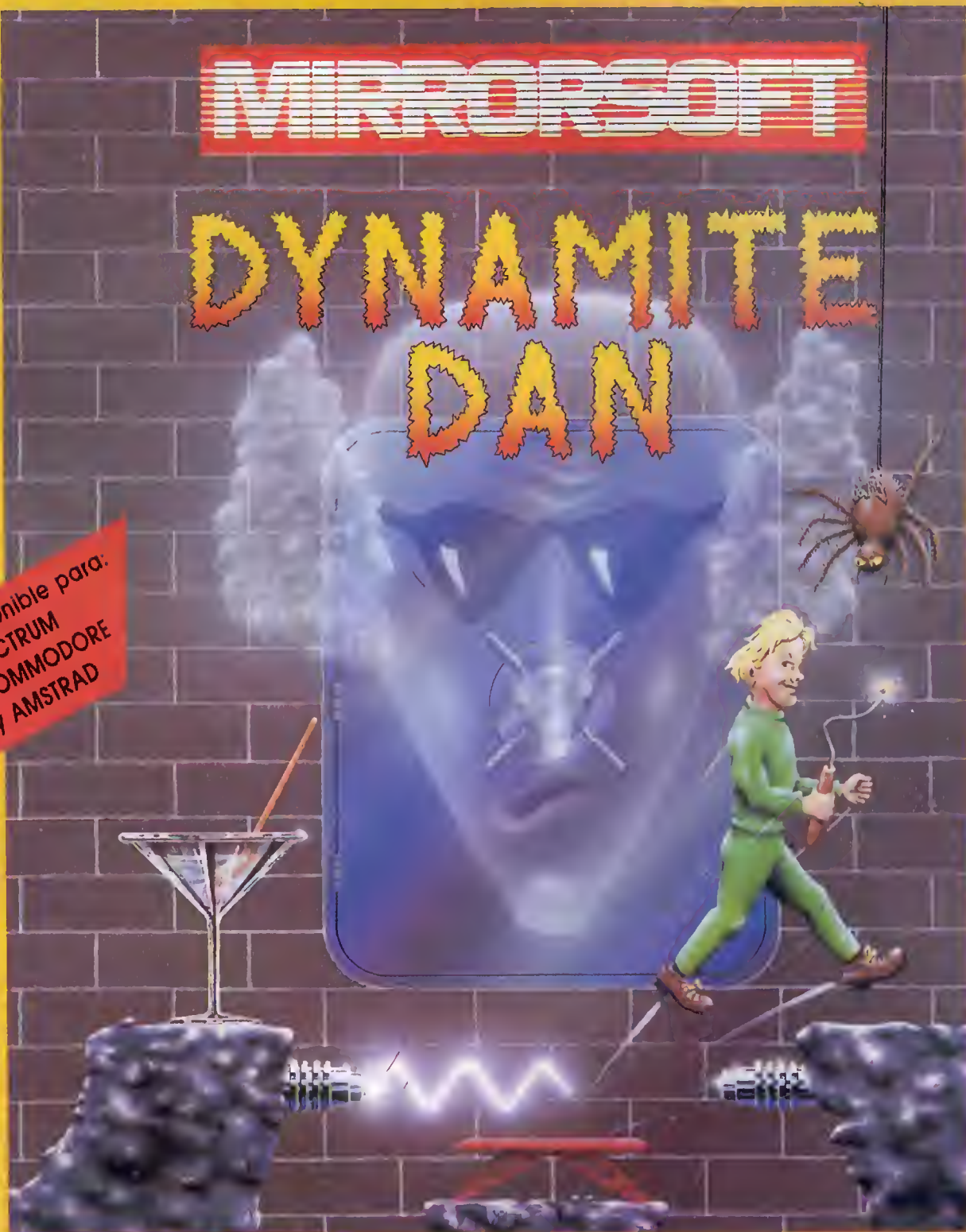
¡¡POR PRIMERA VEZ EN EL MUNDO UN PROGRAMA
DE JUEGOS QUE PUEDES HACER VARIAR A TU MEDIDA CUANTAS VECES QUIERAS!!.

MIRRORSOFT

**DYNAMITE
DAN**

Disponible para:
SPECTRUM
COMMODORE
Y AMSTRAD

2.100 Ptas.



¡No te lo pierdas!

DISTRIBUIDO EN ESPAÑA POR:

círculo de soft

MICROAMIGO S.A.

P.º de la Castellana, 268, 3.º C. 28046-MADRID.
Tel.: (91) 733 25 00

ALIENS ATERRIZANDO

La semana pasada vimos cómo está organizada la memoria de pantalla en Modo 0 con la ayuda de algunos programas sencillitos... En esta ocasión intentaremos dibujar en la pantalla un caracter multicolor por medio de varias rutinas cortas en código máquina.

F. L. Frontán

Antes de nada hagamos un pequeño resumen de lo que conocemos.

La pantalla está organizada en filas de pixels y un caracter sencillo ocupa 32 bytes de memoria, ocho filas de cuatro bytes. El modelo binario de cada byte en la memoria de pantalla contiene información de dos pixels adyacentes horizontalmente.

Distribución de la memoria

Las filas de pixels están asociadas en grupos de ocho descendiendo por la pantalla. Hay 25 de estos grupos en ella —son las líneas que nosotros utilizaremos para situar el cursor del texto con LOCATE y para escribir con PRINT—. La dirección del comienzo de cada fila de pixels de uno de estos grupos está situada &800 bytes después de la correspondiente al comienzo de la fila anterior y cada grupo comienza &50 bytes después que el que tiene encima. La figura 1 nos muestra el Mapa de memoria del ángulo superior izquierdo de la pantalla.

Para dibujar un caracter de tamaño normal todo lo que se necesita es elaborar los datos requeridos y almacenarlos en ocho filas de 4 bytes en algún lugar de la RAM de pantalla. Para que nos sea más fácil, primero imprimiremos un caracter sencillo en una de estas 25 líneas en las que está dividida la pantalla.

Decodificación de un caracter

Para obtener los datos del caracter ejecutamos el programa 1 que

además los coloca a partir de la dirección &9002. Esta información será la que utilicen las rutinas en código máquina contenidas en este artículo.

El programa 2 es un listado en ensamblador de la rutina usada para dibujar el alien. Podemos hacer dos cosas para meter este programa en el **Amstrad**: utilizar un ensamblador para colocar los mnemónicos de las instrucciones o introducir los códigos hexadecimales, de uno en uno, usando el programa 3.

Para ver en acción la rutina de escritura del programa 2 tecleemos:

```
MODE 0: ??:?  
CALL &8000
```

El alien aparecerá en el ángulo superior izquierdo de la pantalla correspondiente a la dirección &C000.

Funcionamiento del programa 2

Primero se cargan el par de registros BC con el número de columnas y filas que estén almacenadas a partir de la dirección &9000, DE con la dirección de los datos, &9002 y HL con la dirección del sitio de la pantalla donde queremos dibujar, &C000. El registro B es el contador de bucle interno, el número de columnas y el C es el del bucle externo, el número de filas.

A continuación salvamos los valores de los contadores y de la dirección de la fila, después el bucle interno la recorre recogiendo el dato contenido en la dirección a la que apunta DE y lo almacena en la dirección señalada por HL.

Incrementamos HL y DE para obtener el siguiente dato y la nueva dirección de la pantalla.

Al final de la primera fila reponemos el valor de la dirección del comienzo y sumamos &800 a HL para obtener la dirección de la siguiente. Recuperamos los valores de los contadores de bucle y decrementamos el número de filas, C.

Si estudia la rutina verá que la anchura del caracter no tiene por qué ser constante. No es imprescindible que tenga cuatro bytes de ancho —la que tiene, precisamente, nuestro alien.

El registro B se carga con la anchura al comienzo del programa y se decrementa cada vez que se hace el bucle interno hasta que llega a ser cero. HL y DE se incrementan para contener las direcciones correctas.

Código máquina



¿Qué pasa con la altura? El alien tiene ocho pixels de alto y se comienza a dibujar en $\&C0$. El bucle externo suma $\&800$ a la dirección contenida en HL cada vez, para obtener lo del comienzo de la siguiente fila así que la dirección de la última es $\&F800$. Supongamos que el carácter no esté exactamente en una línea de texto. Podríamos dibujarlo en el quinto pixel hacia abajo y así quedará en la mitad de una línea y a la mitad de la siguiente.

Aquí encontraremos problemas debido a que el carácter está reportado entre dos grupos de 8 filas. Cuando bajemos a la fila inferior del grupo, sumando $\&800$ a su dirección no obtendremos lo de la fila superior del siguiente grupo de ocho.

Precauciones indispensables

En la figura 1 vemos que la dirección de la primera fila del próximo grupo es $\&C050$, pero $\&F800 + \&800$ nos da cero. Como uno par de registros no puede contener números mayores que $\&FFFF$, si superamos este valor se produce un desbordamiento, se activa el Carry flag y el par de registros comienza otra vez desde cero.

Lo que necesitamos hacer es controlar si se ha producido un desbordamiento y si hay que introducir después un factor de corrección — $\&C050$ —. Si no se ha producido, todo irá correctamente.

El programa 4 es la misma rutina que antes, pero la hemos añadido que compruebe si se produce desbordamiento. Para ensayarlo escribiremos nuestro alien en el quinto pixel hacia abajo, en la dirección $\&E000$.

Las cuatro primeras líneas siguen el tratamiento que hemos explicado en el programa anterior y comienzan en las direcciones $\&E000$, $\&E800$, $\&F000$ y $\&F800$. Cuando sumemos $\&800$ a HL para calcular la dirección de comienzo de la quinta fila se producirá un desbordamiento. Esto activará el Carry flag así que se suma $\&C050$ para corregir el resultado. Observe que realizamos el control en cada fila y no solamente en la quinta.

Teclee de nuevo:

MODE 0: ??:?

CALL &8000

y veremos la mitad de nuestro alien dibujada en la primera línea y la otra mitad en la segunda.

Esta corta rutina imprimirá caracteres multicolores de cualquier tamaño y en el lugar de la pantalla que queramos. No importa si el alien está colocado exactamente en una de las 25 líneas de texto o está repartido entre dos o más, el código máquina lo controla y corrige cuando es necesario. ¡Inténtelo y lo verá! Cargue HL con cualquier valor a partir de &C000, ensamble la rutina de nuevo y llame a &8000.

Las ventajas del código máquina

Es difícil hacerse una idea de si la velocidad de la rutina en código máquina es muy superior a la del Basic cuando se imprime un solo carácter. El programa 5 llena totalmente la pantalla de aliens, y si consideramos que cada alien está formado por varios colores, su rapidez nos deja mudos de asombro.

El par de registros HL se usa para almacenar la dirección y el BC es el contador del bucle. Sus valores se guardan antes de dibujar el alien y se restauran después. La rutina de escritura propiamente dicha está separada del resto, la hemos puesto el LABEL (o *etiqueta*) PRINT por razones evidentes y se la llama como a una subrutina. El par de registros HL se usa para pasar la dirección donde vamos a escribir el carácter.

Creemos que todo lo anterior es suficiente para asimilarlo por esta semana. Pronto veremos cómo se consigue que las cosas se pongan en movimiento.

&C000	&C001	&C002	&C003	&C004	...
&C800	&C801	&C802	&C803	&C804	...
&D000	&D001	&D002	&D003	&D004	...
&D800	&D801	&D802	&D803	&D804	...
&E000	&E001	&E002	&E003	&E004	...
&E800	&E801	&E802	&E803	&E804	...
&F000	&F001	&F002	&F003	&F004	...
&F800	&F801	&F802	&F803	&F804	...
&C050	&C051	&C052	&C053	&C054	...
&C850	&C851	&C852	&C853	&C854	...
...

Figura 1: Mopa de memoria de la esquina superior izquierda de la pantalla en Mode 0

PROGRAMA I

```

10 REM PROGRAMA I
20 POKE &9000,8:POKE &9001,4
30 FOR i=0 TO 31
40 READ J:POKE &9002+i,J
50 NEXT
60 REM ALIEN
70 REM Filas=8/Columnas=4
80 DATA 4,12,12,8,72,148,104,132,72
90 DATA 156,108,132,28,60,60,44,28
100 DATA 32,16,44,4,48,48,8,0,0,4
110 DATA 4,0,0,8

```

PROGRAMA II

```

8000          org #8000
              PROGRAMA II
8000 ED4B0090 LD BC,(&9000)
8004 110290 LD DE,&9002
8007 2100C0 LD HL,&C000
800A          BUCLE1:
800A C5 PUSH BC
800B E5 PUSH HL
800C          BUCLE2:
800C 1A LD A,(DE)
800D 77 LD (HL),A
800E 23 INC HL
800F 13 INC DE
8010 10FA DJNZ BUCLE2
8012 E1 POP HL
8013 01000B LD BC,&800
8016 09 ADD HL,BC
8017 C1 POP BC
8018 0D DEC C
8019 20E9 JR NZ,BUCLE1
801B C9 RET
801C          END

```

ETIQUETAS

BUCLE1 800A BUCLE2 800C

PROGRAMA III

```

10 REM PROGRAMA III
20 a=&8000
30 PRINT HEX$(a):" : "
40 INPUT h$
50 POKE a,VAL("&"+"h$")
60 a=a+1:GOTO 30

```

PROGRAMA IV

```

8000          org #8000
              PROGRAMA IV
8000 ED4B0090 LD BC,(&9000)
8004 110290 LD DE,&9002
8007 2100C0 LD HL,&E000
800A          BUCLE1:
800A C5 PUSH BC
800B E5 PUSH HL
800C          BUCLE2:
800C 1A LD A,(DE)
800D 77 LD (HL),A
800E 23 INC HL
800F 13 INC DE
8010 10FA DJNZ BUCLE2
8012 E1 POP HL
8013 01000B LD BC,&800
8016 09 ADD HL,BC
8017 3004 JR NC,OK
8019 0150C0 LD BC,&C050
801C 09 ADD HL,BC
801D          OK:
801D C1 POP BC
801F 0D DEC C
8019 20E9 JR NZ,BUCLE1
8021 C9 RET
8022          END

```

BUCLE1 800A BUCLE2 800C OK 801D

PROGRAMA V

```

8000          org #8000
              PROGRAMA V
8000 LD HL,&C000
8003 LD BC,500
8006          SIGUE:
8006 E5 PUSH HL
8007 C5 PUSH BC
8008 CD17B0 CALL PRINT
800B C1 POP BC
800C E1 POP HL
800D 23 INC HL
800E 23 INC HL
800F 23 INC HL
8010 23 INC HL
8011 08 DEC BC
8012 78 LD A,B
8013 B1 OR C
8014 20F0 JR NZ,SIGUE
8016 C9 RET
8017          PRINT:
8017 ED4B0090 LD BC,(&9000)
801B 110290 LD DE,&9002
801E          BUCLE1:
801E C5 PUSH BC
801F E5 PUSH HL
8020          BUCLE2:
8020 1A LD A,(DE)
8021 77 LD (HL),A
8022 23 INC HL
8023 13 INC DE
8024 10FA DJNZ BUCLE2
8026 E1 POP HL
8027 01000B LD BC,&800
802A 09 ADD HL,BC
802B 3004 JR NC,OK
802D 0150C0 LD BC,&C050
8030 09 ADD HL,BC
8031          OK:
8031 C1 POP BC
8032 0D DEC C
8033 20E9 JR NZ,BUCLE1
8035 C9 RET
8036          END

```

ETIQUETAS

BUCLE1 801E BUCLE2 802D OK 8031
PRINT 8017 SIGUE 8006



COMPUTIQUE

Te da más

AMSTRAD
GARANTIA ESPAÑA

64.900Ptas.

Amstrad 464 f.
verde



Al comprar tu Amstrad te regalamos

- Estuche con ocho programas originales
 - Fruit Machine
 - Procesador texto
 - Almirante Graf
 - Oh Mummy
 - Plaga Galáctica
 - Amsdraw
 - Laberinto Sultan
 - Animal, Vegetal, Mineral
- Joystick Gunshot I
- Un estupendo libro de Basic
- Los cuatro mejores programas:
 - Decathlon
 - Sabrewulf
 - Jet Set Willy
 - Beach-Head
- Guía de referencia del programador
- y además te obsequiamos con un curso de introducción al Basic.

VENTA A PLAZOS HASTA 36 MESES



Nuevo Amstrad CPC6128: 109.500 ptas. (F. Verde)

COMPUTIQUE

Servimos a tiendas
Abrimos sábados por la tarde

Embajadores, 90 Tfno. 2270980
28012 Madrid

Sin duda alguna

A través de esta sección se pretende resolver, en la medida de lo posible, todas las posibles dudas que «**atormenten**» a todas las personas interesadas en el mundo del AMSTRAD, sean o no poseedores de uno y, si lo son, se encuentren en cualquier nivel de destreza en su manejo.

Semanalmente, aparecen en estas páginas las consultas de la mayor cantidad de usuarios posible; ello redundará en un mejor servicio y en un contacto más estrecho entre todos nosotros a través de la revista.

SIN DUDA ALGUNA está abierta a todos.

Muy señores míos:

Me voy a comprar un **Amstrad CPC 464** y la pregunta es: ¿Se le puede poner una unidad de disco al **Amstrad CPC 464** y ser compatibles los discos que salgan para el CPC 664?

Un saludo.

Francisco Fernández (Málaga)

Al CPC 464 puedes colocarle una unidad de disco. En principio, prácticamente la totalidad de software en disco, sobre todo bajo CP/M, correrá en ambos ordenadores, 464 y 664.

Me dirijo a ustedes para consultarles dos dudas. Una es referente a un juego publicado por esta revista en el número dos: «Egg Blitz». Este programa lo tecleé en mi ordenador (un CPC 464) y éste me indicaba «Memory full». Les ruego miren si esto es debido a alguna errata de publicación o a algún error de programación.

La otra duda es sobre la posibilidad de ampliación de memoria del **Amstrad CPC 464**. ¿Es posible? Si es así, ¿cómo?

¡Ah!, uno terdora dudo. ¿Podrían explicarme los esquemas de períodos de tono de los manuales del CPC 464? y ¿cómo se pueden generar las notas de la escola musical con «sound»?

Ahora ya sólo me resta despedirme. Gracias por escucharme.

Ramón J. Pachades

Respecto a tu primera cuestión, suponemos que intentaste cargar el programa en un 464 con unidad de disco (el mensaje MEMORY FULL así lo indica), porque de lo contrario funciona perfectamente.

Donde sí tendrás problemas con este programa es en un 664, por la razón anteriormente expuesta, el disco.

La posibilidad de ampliar la memoria del 464 es algo que todavía no está claro: teóricamente es posible, paginando memoria, pero que sepamos no hay nada comercial disponible.

Tengo en mente cambiar mi CPC-464 por un CPC 6128 y he oído rumores de que los programas en cinta comerciales e incluso los vuestros de la Serie Oro, no son compatibles con este ordenador.

¿Hay algo de cierto en esto?

Gracias.

Nota: Enhorabuena, os está saliendo muy bien la revista.

Mario Durán (Madrid)

Nuestros programas normalmente son compatibles con todos los **Amstrad** y, en caso contrario, se indica en la revista con toda claridad a qué ordenadores van dirigidos.

En el caso de soft comercial, existen programas que sólo funcionan en el 464.

Normalmente, los que corren en el 664, también lo hacen en el 6128.

Soy el casi arrepentido poseedor de un ordenador **Amstrad CPC 664**. Como no he podido obtener en caso comercial alguna, ninguno de los periféricos de este ordenador ni ninguno de los programas de gestión en disco que **Amstrad** insiste en anunciar, opté por entretener la ya larga espera aprovechando los juegos que tiene a su disposición el CPC 464. Utilicé para ello un cable de conexión al magnetófono, hecho por profesionales a propósito, ya que el cable de la casa tampoco existe. Lo decepción ha superado todas las previsiones.

Lo mayor parte de los juegos probados no se graban. En estos casos se exhibe con unanimidad el mensaje: «Memory full in ...» (En lugar de la línea de puntos aparece un número.) En el libro de instrucciones no se explica nada. ¿Saben ustedes qué significa este mensaje? ¿Es posible superar este problema? ¿Hasta qué punto son compatibles los juegos del CPC 464 en el CPC 664?

Esteban Padros (Barcelona)

Que nosotros sepamos, Indescomp vende un cable de cassette para permitir que el 664 pueda aprovecharse del software existente en cinta.

Por otra parte, algunos de los programas que existen para el 464 no son compatibles para el 664. Normalmente, esto sucede con los primeros programas que salieron para **Amstrad**, cuando el 664 aún no estaba disponible y se irá resolviendo con el tiempo.

Una casa de software respetuosa con sus clientes, lo menos que puede hacer es indicar en la cinta con qué máquina es compatible su programa. Como desgraciadamente no es así, la única solución es exigir que prueben el programa que piensas adquirir en un 664.

NL New Line
GABINETE DE INFORMÁTICA

- **Clases de Informática sobre AMSTRAD**
EN GRUPOS O INDIVIDUALES
- **Ordenadores AMSTRAD y periféricos**
Los mejores precios
- **Software a la medida**

ZURBANO, 4 ☎ 410 47 63
28010 MADRID

K-BITS

Ordenadores personales y de gestión
Aplicaciones para arquitectura

- | | |
|----------------|---------------|
| • Amstrad | • Dragón |
| • Sinclair | • Impresoras |
| • Commodore | • Monitores |
| • Philips | • Periféricos |
| • Canon | • Libros |
| • Spectravideo | • Revistas |

Servimos a provincias

C/ Barquillo, 15.
Teléfono: (91) 232 57 37. Madrid.

¿Cómo conseguir que una serie de datos, sean utilizados por varios programas sin tener que teclearlos en cada uno de ellos? La solución, es crear un fichero de datos, independiente del programa, que se almacena en cinta o disco y del cual podemos recuperarlos cuando nos interesa.

En esta ocasión Análisis estudia como grabar datos en un archivo en cinta o disco y como leerlos después.

10,20 Los REM de costumbre.

30 Escribe un mensaje en pantalla, que nos indica que los datos se están grabando.

Su inclusión, se debe a que en los sistemas de disco esta grabación, se realiza sin mensaje alguno y se podría pensar que no está ocurriendo nada.

En los sistemas de cinta, es innecesario por la aparición del mensaje Press REC and PLAY then any key.

50 Efectúa la apertura de una vía de transmisión de datos desde el programa a la cinta o al disco, permitiendo que éstos sean almacenados en un fichero, en este caso le hemos dado el nombre de archivo.

60-90 Ciclo FOR...NEXT, que se encarga de leer los datos y mandarlos al archivo.

Una vez comprendida el programa sería interesante, hacer que éste se repita dos o tres veces y observar los resultados.

70 Lee los datos almacenados en las variables; nombre\$ y número.

En nuestro caso la información viene de una línea DATA, pero esta podría realizarse también desde otros cauces, como por ejemplo el teclado.

80 WRITE #9 hace que la información almacenada en las anteriores variables, sea enviada a disco o a cassette.

La primera vez que rota el ciclo les corresponde a Pepe y 1, la segunda; a Luis y 2, la tercera; a Carlos y 3.

90 El ciclo concluye después de tres vueltas, leyendo toda la información de la línea DATA.

100 Envía una cadena que será almacenada en el fichero de salida, ésta indica el final del archivo.

110 Cierra el canal de salida.

130-140 Provoca una pausa entre el almacenamiento y la lectura de datos.

En este momento, los datos están almacenados en un fichero llamado archivo.

160-230 Lee los datos del fichero y los copia en pantalla.

160 Creo una vía de comunicación entre el programa y el sistema de almacenamiento, para leer el fichero archivo. En el caso de utilizar cassette, rebobinar la cinta.

170-200 Conforman el ciclo FOR...NEXT, que efectúa la lectura de los datos.

180 INPUT #9 toma información de disco o cassette, almacenándola en las variables nombre\$, número.

190 Visualizo en pantalla los datos tomados del archivo.

210 Efectúa la lectura del final del fichero y la almaceno en final\$.

220 Escribe el mensaje «Final del archivo».

230 Cierra la vía de comunicación establecido en la línea.

160

250 Contiene los datos que vamos a transferir al fichero.



Mercado común

Con el objeto de fomentar las relaciones entre los usuarios de AMSTRAD, **MERCADO COMUN** te ofrece sus páginas para publicar los pequeños anuncios que relacionadas con el ordenador y su mundo se ajusten al formato indicado a continuación.

En **MERCADO COMUN** tienen cobida, anuncios de ventas, compras, clubs de usuarios de AMSTRAD, programadores, y en general cualquier clase de anuncio que pueda servir de utilidad a nuestros lectores.

Envíanos tu anuncio mecanografiado a: **HOBBY PRESS, S.A.**

AMSTRAD SEMANAL.

Apartado de correos 54.062

28080 MADRID

¡ABSTENERSE PIRATAS!

AMSTRAD Semanal comunica a todos sus lectores la apertura de una nueva sección dedicada a recoger las mejores ideas que exploten al máximo las posibilidades del ordenador, materializadas en programas claros y cortos (*máximo 25 líneas*). Los mejores de entre todos ellos serán publicados con el nombre de su autor en la revista, recibiendo como premio, gratuitamente en su domicilio los cuatro primeros números de nuestra cinta mensual. Los programas enviados deberán incluir:

- **Cinta de cassette con el programa o programas grabados.**
- **Explicación detallada del funcionamiento y propósito del programa, mecanografiado a 2 espacios o con letra clara.**

Es imprescindible indicar en el sobre claramente: **AMSTRAD IDEAS.**

La dirección es:

Editorial Hobby Press, S. A.

La Granja, s/n.

Polígono Industrial de Alcobendas.

Madrid.

Amstrad Ideas

Vendo Commodore 64. Incluya Datassette C2N, joystick Quick Shot II, guía del usuario, Curso de Basic (2 libros), numerosas revistas especializadas, y un excelente surtido de juegos (*One on One, Decathlon, Beach Head, etc.*). Todo ella por 49.000 ptas.

Interesados llamar, de 14 a 16,30 al (972) 31 53 86. O bien, escribir a: Miquel Gasull Buenaventura. Alvaro de Bazán, 2. Palamós (*Girona*).

Vendo «Jaystick» Quickshot I (sin estrenar), «Cadename Mat», más «Gremlins» (en castellano), más «Fighter Pilot», más «Decathlon» o «World Cup Football», toda por 3.500 ptas. Gastos de envía por mi cuenta. Instrucciones completas y en castellano. Escribir a Manuel Angel Sánchez Casta. Amílcar Barca, 29 - 2.º D. 11009-Cádiz.

MICRO-1

Jorge Juan, 116. 28028 Madrid
Tel. (91) 233 07 35-274 53 80

Regala de bienvenida al club de Software MICRO-1: 1 bolígrafo de acera con reloj de cuarza incorporada.

match day	1.975	dragontorc	1.875	raid o moscow	1.975
southern belle	1.975	dummy run	1.975	jump jet	2.595
death pit	1.975	combat lynx	1.925	beach head	2.395
decathlon	1.975	exploding fist	2.095	basketball	1.950
gremlins	1.975	racky	1.925	highway encout	1.975

¡Increíbles precios para tu AMSTRAD 464 y 128!
(llámanos, y te asombrarás)

Lápiz Optico DKTRONICS 4.850

Tapa de metacrilato transparente para tu
AMSTRAD 2.450

Impresoras: ¡20% Dta. sobre P.V.P.!

Diskette 3": 1.050
Cinta-15: 85

El pedido te lo enviamos **URGENTEMENTE** contra-reembolso SIN NINGUN GASTO DE ENVIO, LLAMANDO a los teléfonos (91) 233 07 35-274 53 80 o escribiendo a MICRO-1. Jorge Juan, 116. 28028 Madrid.

SOLICITUD DE CINTAS DE PROGRAMAS Y NUMEROS ATRASADOS

☐ Deseo recibir en mi domicilio, al precio de **675 ptas.** cada una, las siguientes cintas con los programas publicados por **Microhobby AMSTRAD** Marco con una (X) la(s) cinta(s) que deseo:

- ☐ Cinta n.º 1 (contiene programas publicados en revistas 1 al 4 inclusive)
- ☐ Cinta n.º 2 (contiene programas publicados en revistas 5 al 8 inclusive)
- ☐ Cinta n.º 3 (contiene programas publicados en revistas 9 al 12 inclusive)
- ☐ Cinta n.º 4 (contiene programas publicados en revistas 13 al 16 inclusive)
- ☐ Cinta n.º 5 (contiene programas publicados en revistas 17 al 20 inclusive)

☐ Deseo recibir en mi domicilio, al precio de **150 ptas.** cada uno, los siguientes números atrasados de **Microhobby AMSTRAD**

Nota: Por razones administrativas, no podemos admitir solicitudes de envío de cintas o números sueltos con pago contra reembolso o Tarjeta de Crédito. Por favor, envía talón por el importe o giro postal.
Si la forma de pago elegida es talón bancario, remítelo junto con este cupón en un sobre cerrado a la misma dirección.
Las cintas de programas se editan una cada mes. Si solicitas varias las recibirás sucesivamente, conforme sean editadas. No se cobran gastos de envío por las cintas y números sueltos.

NOMBRE _____ EDAD _____
 APELLIDOS _____
 DOMICILIO _____
 CIUDAD _____ PROVINCIA _____
 C. POSTAL _____ TELEFONO _____ PROFESION _____

Marco con una (X) en el casillero correspondiente la forma de pago que más me conviene.

☐ Talón bancario adjunto a nombre HOBBY PRESS, S. A. ☐ Giro Postal N.º _____

Firma: _____



Producido en exclusiva para España por:

ACE

Actividades Comerciales Electrónicas, S.A.
 Carrereta 100 - Tel. 325 10 50* 09115 Barcelona. Télex 93133 ACE E E

YA DISPONIBLE EN

El Corte Inglés

Y EN TODAS LAS
 TIENDAS ESPECIALIZADAS

Franqueo
Postal

ción
MER
nas p
que l
munc
conti
En
anun
usual
y en
que l
lector
En
o: H

AMSTRAD SEMANAL.

Apartado de correos 54.062
28080 MADRID

¡ABSTENERSE PIRATAS!

especializadas, y un excelente
surtido de juegos (*One on One*,
Decathlon, *Beach Head*, etc.).
Todo ello por 49.000 ptas.
Interesados llamar, de 14 a
16,30 al (972) 31 53 86. O bien,
escribir a: Miquel Gasull
Buenaventura. Alvaro de Bazán,
2. Palamós (*Girona*).

HOBBY PRESS, S.A.

Apartado de Correos

n.º 54.062 (Apartados Altos)

MADRID

«World Cup Football», todo por
3.500 ptas. Gastos de envío por
mi cuenta. Instrucciones completas
y en castellano. Escribir a Manuel
Angel Sánchez Costa. Amílcar
Barca, 29 - 2.º D. 11009-Cádiz.

MICRO-1

Jorge Juan, 116. 28028 Madrid
Tel. (91) 233 07 35-274 53 80

Regalo de bienvenida al club de Software MICRO-1: 1 bolígrafo de acero con reloj de cuarzo incorporado.

match day	1.975	dragontorc	1.875	raid o moscow	1.975
southern belle	1.975	dummy run	1.975	jump jet	2.595
death pit	1.975	combot lynx	1.925	beach head	2.395
decathlon	1.975	exploding fist	2.095	basketball	1.950
gremlins	1.975	rocky	1.925	highway encout	1.975

¡Increíbles precios para tu AMSTRAD 464 y 128!
(llámanos, y te asombrarás)

Tapa de metacrilato transparente para tu
AMSTRAD 2.450

Lápiz Optico DKTRONICS 4.850

Impresoras: ¡20% Dto. sobre P.V.P.!

Diskette 3": 1.050
Cinta-15: 85

El pedido te lo enviamos URGENTEMENTE contra-reembolso SIN NINGUN GASTO DE ENVIO, LLAMANDO
o los teléfonos (91) 233 07 35-274 53 80 o escribiendo a MICRO-1. Jorge Juan, 116. 28028 Madrid.

¡SUSCRIBETE A Microhobby AMSTRAD Y AHORRA 1.600 PTAS.!!

3-D VOICE CHESS

Ajedrez tridimensional con voz en castellano
Amstrad CPC 464, CPC 664 y CPC 6128

P.V.P.

2.100.- (cinta)

3.300.- (disco)



PUBLISHED
**DEEP THOUGHT
SOFTWARE**

DISTRIBUTED

CD Software



• Brillantes gráficos 3-D
(colores reales)
• Niveles de juego
(de principantes a expertos)
• Rápidas respuestas desde 5 segundos

Producido en exclusiva para España por:

ACE

Actividades Comerciales Electrónicas, S.A.

Taragona 100 - Tel. 325 10 58* 08015 Barcelona. Telex 93133 ACE E E

YA DISPONIBLE EN



Y EN TODAS LAS
TIENDAS ESPECIALIZADAS

Rocky



AZPIRI

DINAMIC

Simulación tridimensional de un combate de boxeo. Lucha contra CIMBELIN, TED MATARE, JANSEN SINO y FIGHTER BULL para alzarlo con el título mundial.

SPECTRUM 48 K 1.800 ptas.
AMSTRAD CPC 464. 2.100 ptas.

Rocky



Servimos a tiendas y almacenes telf. (91) 447 34 10 Telex 22542 JAGA E

Pedidos contrareembolso (Sin G/E y adhesivos regalo) a:

"MANSION" DINAMIC: C/ Tilos, 2,21 MONTEPRINCIPE, BOADILLA DEL MONTE, MADRID TELS. (91) 715 00 67